Educational Services

digital™

EY-9222E-SA-0001

VAXcluster Maintenance
Student Workbook

Volume 1

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

The following are trademarks of Digital Equipment Corporation:

CI
DEC
DECnet
DECsystem-10
DECSYSTEM-20
HSC
IAS
J-11
LA12/50/70
MASSBUS
PDP
PDT
RSTS
RSX
RX
UNIBUS
VAX
VAXcluster
VAXsim
VMS
digital

CONTENTS

Volume 1

INTRODUCTION TO VAXclusters

Introduction to VAXclusters

Lesson Introduction

The VAXcluster is a highly-integrated organization of VAX/VMS systems that communicate over a high-speed communications path.

It uses software which is already part of the VMS operating system. Therefore, no additional software products are required to install software on a VAXcluster.

The VAXcluster provides an environment in which all of the VAX systems function as a single system. They share the same system resources as well as having a single security and management domain.

This lesson will give an overview of the parts that make up the cluster as well as showing its relation to other types of multiple processor systems.

Lesson Objectives

1. Identify the relationship between a VAXcluster and other coupled computer systems.
2. Describe the function of a VAXcluster and list the advantages.
3. Identify the major hardware components of the VAXcluster and describe the services they perform.
4. Identify the major software components of the VAXcluster and describe the function of each.

Lesson Outline

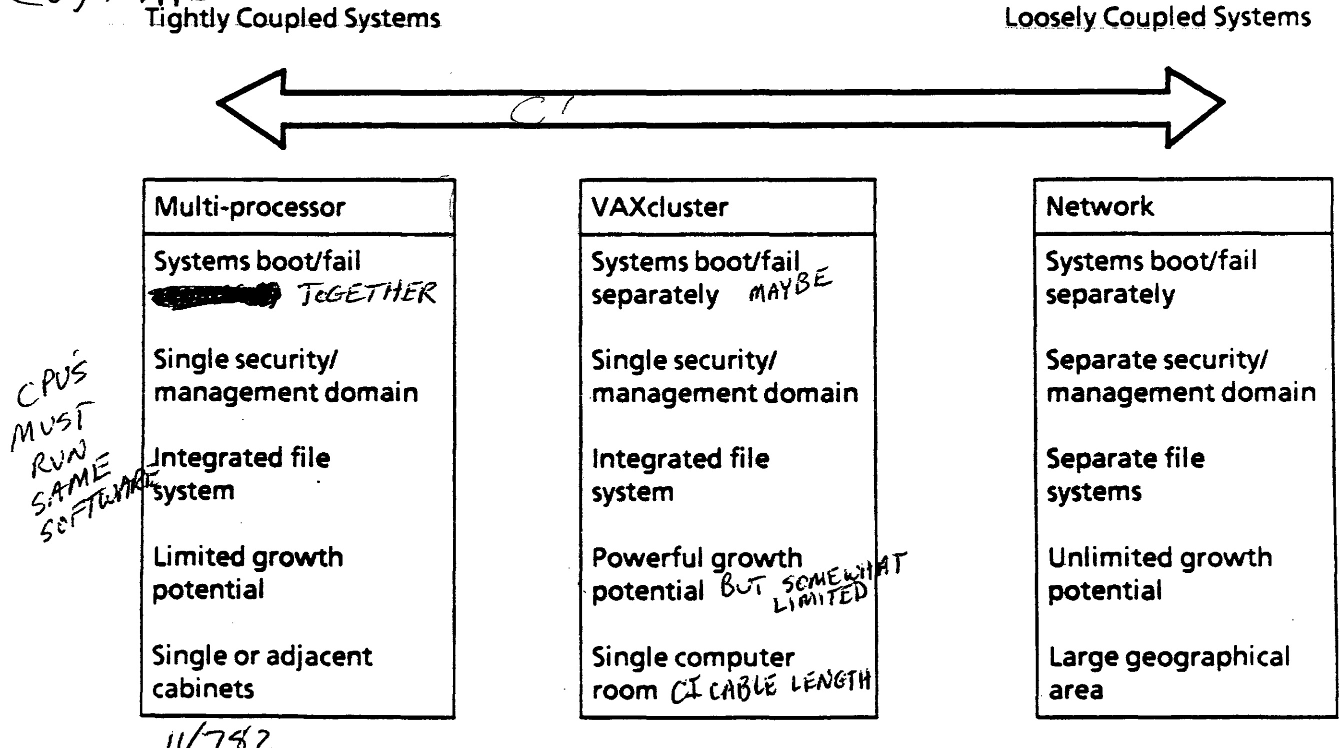
I. VAXcluster Introduction
II. Hardware Components
III. Software Components

Introduction

A VAXcluster is a collection of VAX systems connected by a Computer Interconnect (CI) Bus.

Cluster Definitions

CI Bus	A Computer Interconnect Bus.
CI Cluster	A collection of VAX computers connected by the CI Bus.
VAXcluster	A type of CI Cluster where all the CPUs are running VMS.
CI Port	A type of interface on the CI Bus.
HSC	A mass storage controller for RA disks and TA tape drives.
Star Coupler	A central connection for the CI Bus.
LAVC [illegible]	LOCAL AREA VAXCLUSTER ALSO CALLED (NI) NETWORK INTERCONNECT
(V5) → MI	MIXED INTERCONNECT LAVC & CI



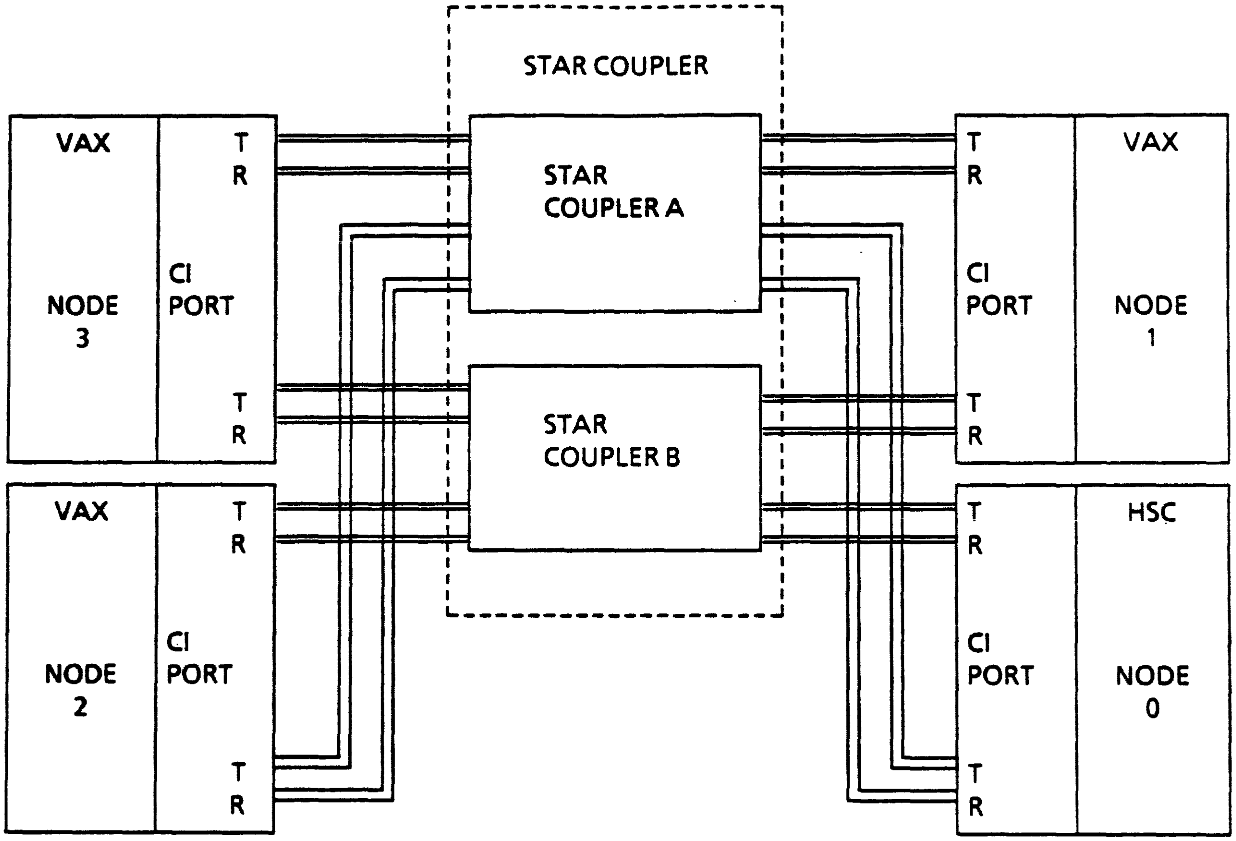
Multi-processor	VAXcluster	Network
Systems boot/fail [redacted] TOGETHER	Systems boot/fail separately MAYBE	Systems boot/fail separately
Single security/ management domain	Single security/ management domain	Separate security/ management domain
Integrated file system	Integrated file system	Separate file systems
Limited growth potential	Powerful growth potential BUT SOMEWHAT LIMITED	Unlimited growth potential
Single or adjacent cabinets	Single computer room CI CABLE LENGTH	Large geographical area

CPUS MUST RUN SAME SOFTWARE

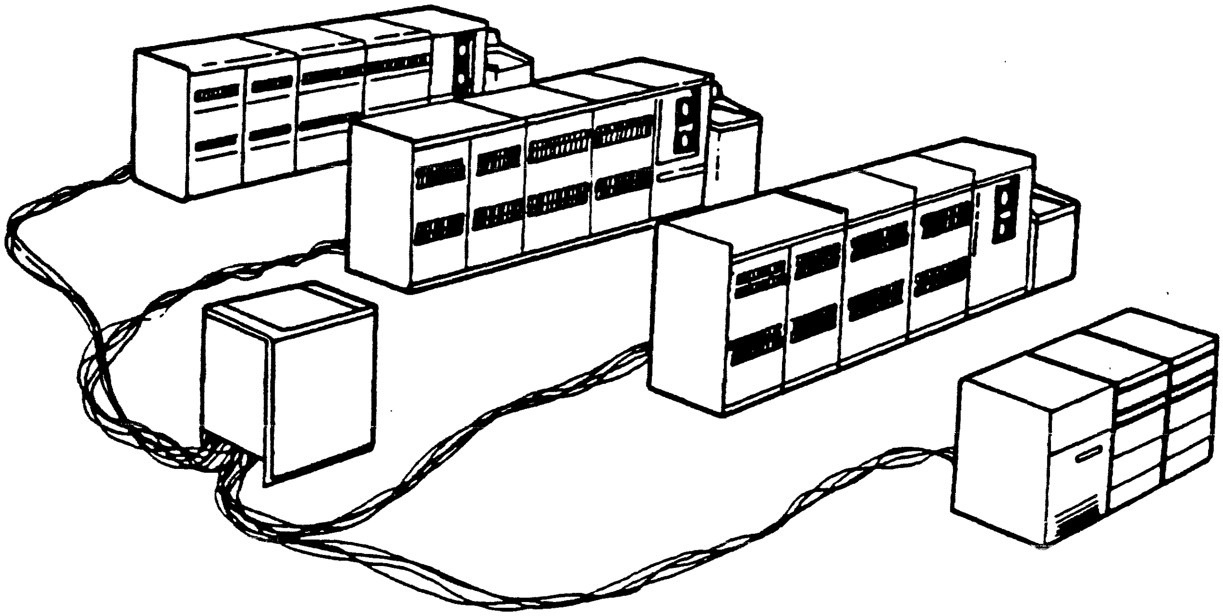
11/782
8800
8350
6000
9000

ZK-1344-83

VAXclusters and Other Multiprocessors



CI Bus Detail Layout



Four Node VAXcluster Physical Layout

Hardware Components of a VAXcluster

- VAX (8XXX -7xx)
  - VMS is running and is extended across the VAXcluster.
  - Active node: a member of the VAXcluster. HSC= NON MEMBER - NO HAS VOTES
  - All active nodes coordinate their actions with respect to mass storage utilization.
  - A VAXcluster should be treated as a single system.
- HSC (Hierarchical Storage Controller)
  - Controls RA-type or TA-type tapes. DSA OPTICALS + SOLID STATE DRIVES.
  - Optimizes access to storage devices. SMART SEEKS (RE ARRANGED TO BE EFFICIENT)
  - Passive node -- not a member of the VAXcluster.
  - Makes all storage devices look like a series of good logical blocks to VMS.
  - Contains resident diagnostic and utility programs.
- CI Bus
  - High-speed (70 megabit), multi-access bus.
  - Coaxial cables for transmit and receive.
  - Dual signal paths.
- Star Coupler (SC008) -- central connection point for CI nodes.
- CI Intelligent I/O Port
  - Connects VAX systems to the coaxial CI Bus.
  - Supports serialized packet communications on the CI Bus.

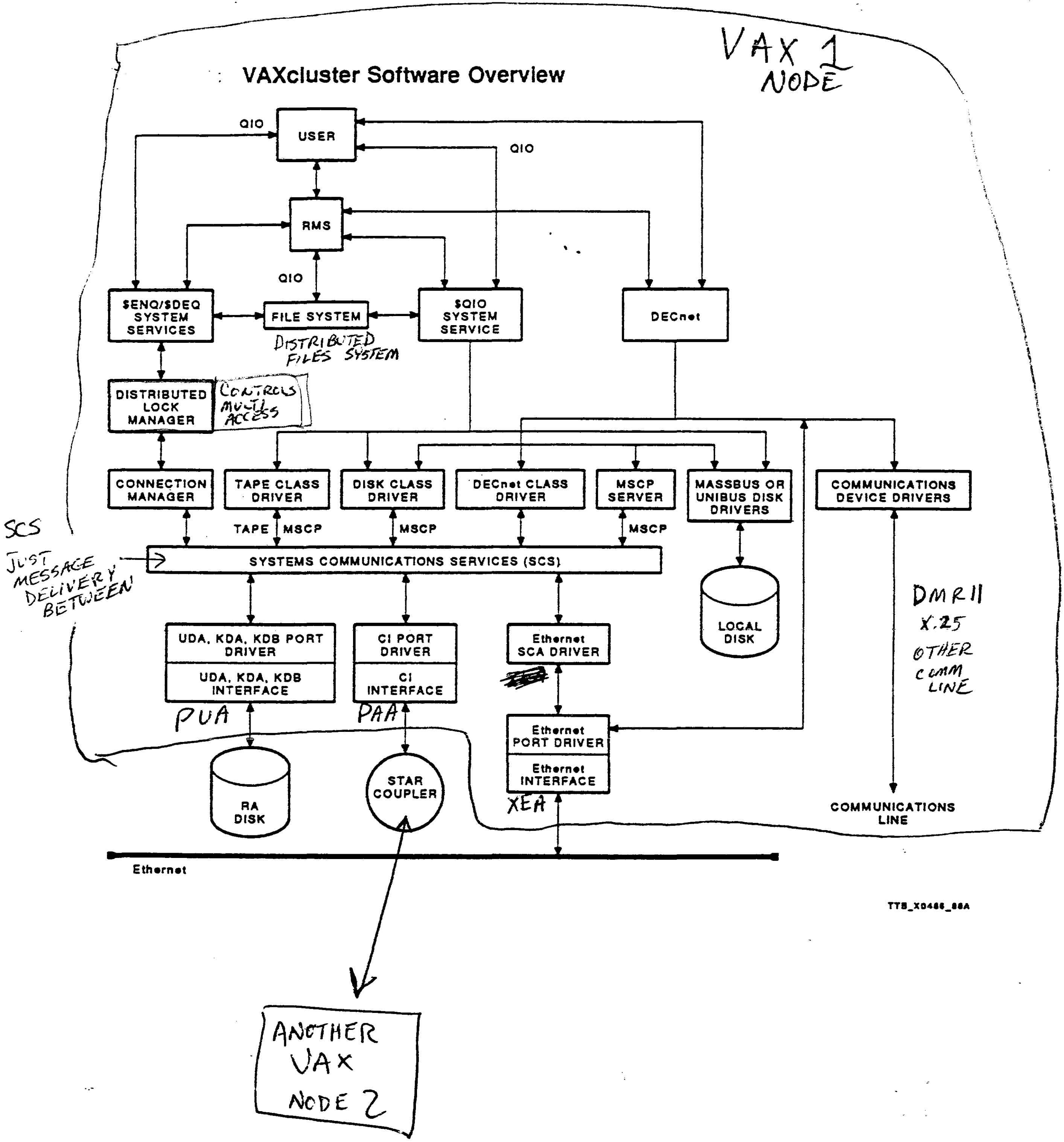
SERIAL TRANSMISSION
OF PACKETS ON BUS

Software Components of a VAXcluster

- Distributed File System
  - Used in conjunction with Record Management Services (RMS) to allow the same access to disks on a cluster-wide basis as is allowed on a single system.
  - The Distributed File System and RMS use the lock manager to coordinate cluster-wide file access.
- Distributed Lock Manager
  - Uses the Connection Manager and System Communication Services (SCS) to communicate information over the CI Bus.
  - Allows cluster-wide synchronization of access to shared resources.
  - Used by the file system, RMS, and the job controller.
- Distributed Job Controller
  - Uses the Distributed Lock Manager to signal other VAX nodes to examine the batch and print queues for jobs to be processed.
  - Permits users to submit batch/print jobs to queues that execute on any node in the cluster.
- Connection Manager
  - Runs on each processor in a cluster.
  - Conducts cluster state changes (joining/leaving cluster, quorum check).
  - Uses SCS to communicate across the CI.
- MSCP Server -- Allows MASSBUS and UNIBUS disks to be made available to other nodes in the cluster by software emulation of an HSC50/70.
- System Communication Services (SCS) -- Software layer that implements internode communications, according to Digital's System Communication Architecture. The SCS layer interfaces the System Application Software (SYSAPS) with the CI Bus.

ALLOWS LOCAL DISKS TO BE AVAILABLE TO CLUSTER

VAXcluster Software Overview



THE COMPUTER INTERCONNECT

The Computer Interconnect

Lesson Introduction

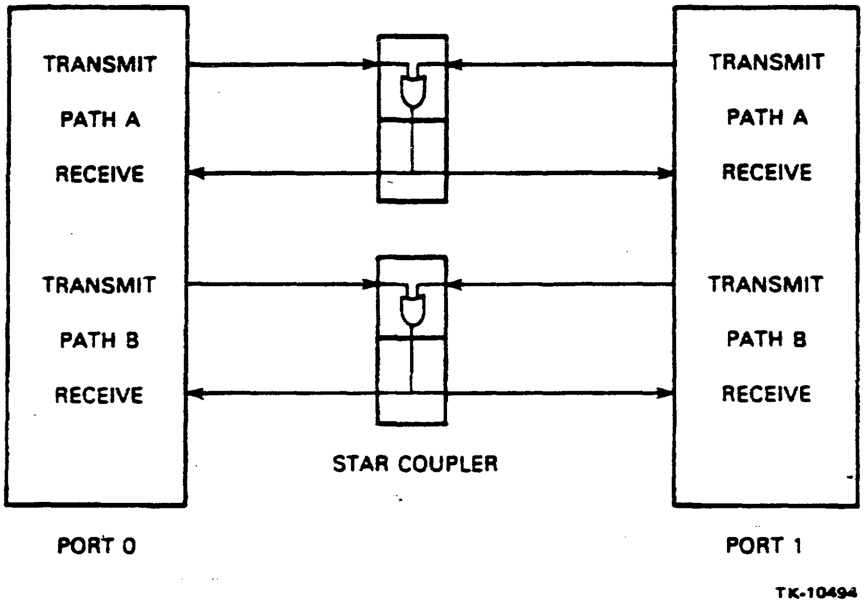
The CI Bus is the communications path (or link) between the nodes in a cluster. The CI has its own protocol, which simply embeds any information to be passed between the nodes inside of a CI Informational Packet. This section is a discussion of communication over the CI Bus as well as a look at the physical connections themselves.

Lesson Objectives

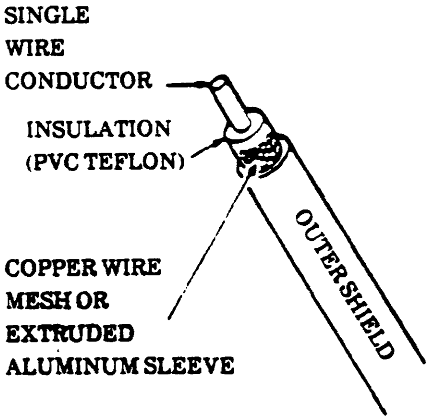
1. Describe the physical characteristics of the CI Bus.
2. Describe how communication is implemented over the CI Bus.
3. Define each of the elements that make up a CI Packet.
4. Describe CI Bus transmission, reception, and arbitration.

Lesson Outline

I. Physical Characteristics
II. Packet Format
III. CI Transmission
IV. Virtual Circuit vs. Software Connection
V. CI Packet Body
VI. CI Data Flow



CI Bus Providing Dual Paths to Each Port



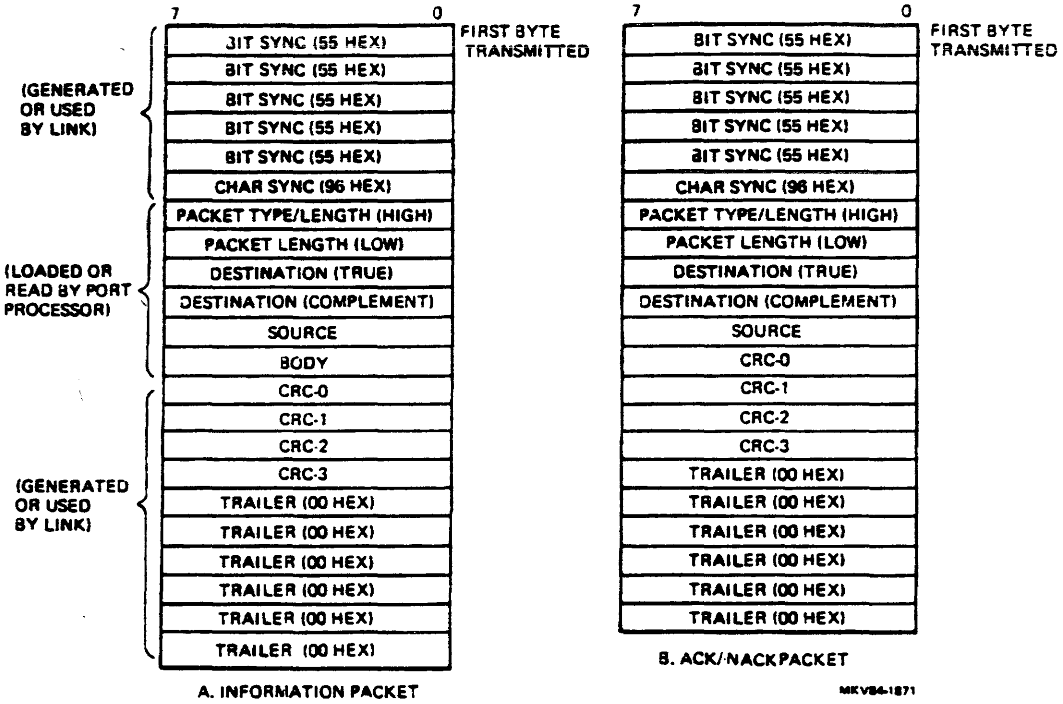
Coaxial Cable

Packet Format Breakdown

- Bit synchronization
  - Turns on the carrier detect circuitry.
  - Synchronizes the Manchester decoder.
  - Consists of alternating ones and zeroes.
- Character synchronization -- used by serial-to-parallel converter to frame the incoming bits into bytes.
- Packet type/length (high)

Bit 7	6	Packet type
0	0	INFO
0	1	not used
1	0	NAK
1	1	ACK

  - Bits 5:4 are always zero.
  - Bits 3:0 are upper four bits of the 12-bit packet length (zero for ACK/NAK).
- Packet length (low) -- used only for information packets, contains the lower eight bits of the packet length field.
- Destination -- matches the address set in the link board switches on the destination node
- Destination (complement) -- matches the complement of the address set in the link board switches on the destination node.
- Source -- matches the address set in the link board switches on the source node.
- Body -- the message (used only for information packets).
- CRC -- a four-byte CRC code for message.
- Trailer -- six bytes of zeroes.



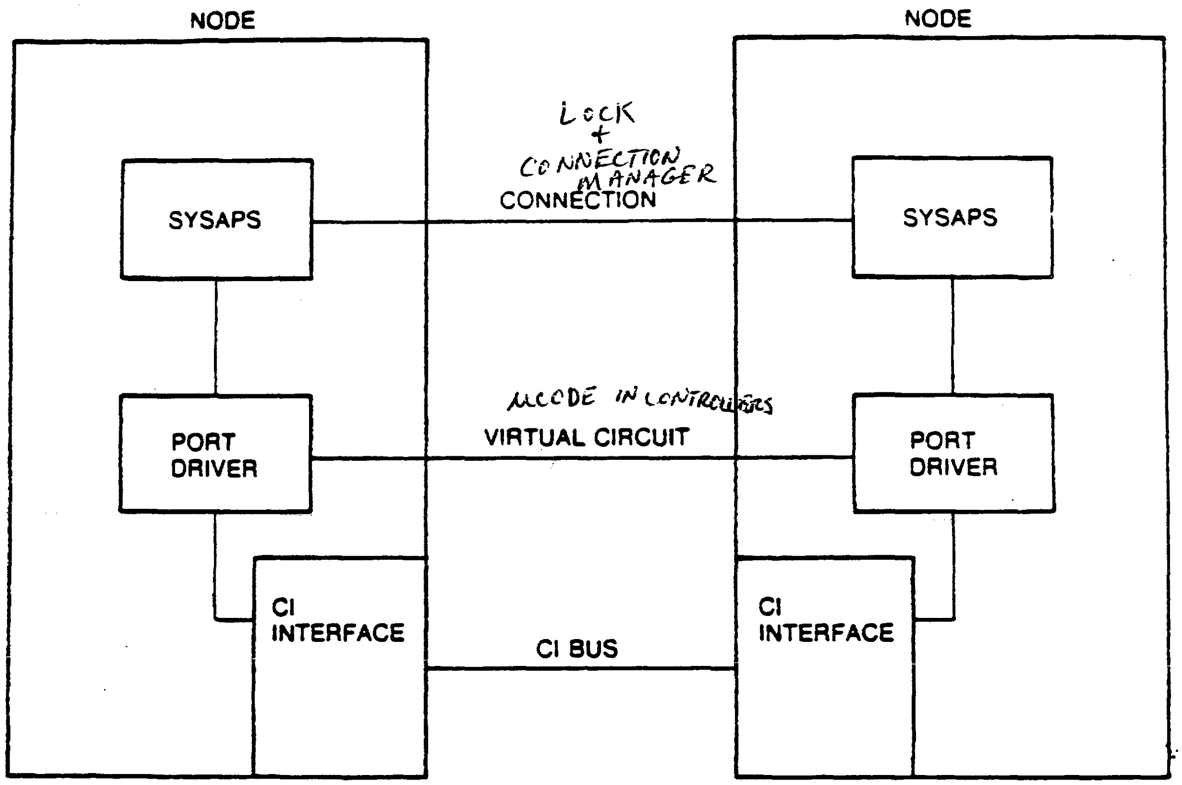
Packet Formats

Transmission on the CI Bus

- Path selection for transmission
  - Normally, the path is selected at random by the microcode.
  - Periodically, a specific path is selected by the software (configuration poller) in order to determine cluster configuration.
- Responses (expected immediately on the same path)
  - ACK = successful reception.
  - NAK = unsuccessful reception because buffer is full.
  - NO PACKET = TIME OUT No response (NO__RSP) = unsuccessful reception because of collision, CRC error, and so forth.
- Retries (microcode-driven)
  - 64 NO__RSPs are allowed per path.
  - 128 NAKs are allowed per path (a NAK resets the NO__RSP counter).
  - No error is logged until the retry count is exhausted.
  - The port switches paths and starts over when the count is exceeded for the selected path.
  - The node closes the virtual circuit (port-to-port logical connection) when the count is exceeded for both paths.

Cluster Node-to-Node Links

The diagram below illustrates the terms used to describe the various types of links that exist between nodes. Note that a "connection" and a "virtual circuit" are LOGICAL links while the CI Bus is the PHYSICAL link.



Node-to-Node Physical and Logical Link

Connection — A System Communication Architecture (SCA) logical path by which two processes running in different nodes communicate. Connections are multipexed within a single virtual circuit between nodes.

Virtual Circuit — A logical connection path between two nodes in a cluster. The message traffic of communication processes is channeled (multiplexed) through this circuit.

Components of the CI Packet Body

- The body of a CI Information Packet is usually of three types:

  Sequential message — Typically an MSCP command; delivery of sequential message is guaranteed.

  Datagram — An intersystem communication message the delivery of which is not guaranteed; for example, an error log message. DECNET MESSAGES ALSO

  Block Transfer — Block data going to/from a storage device;~~or DECnet data~~ between systems.

- An MSCP command (sequential-type message) is not considered complete until an MSCP response from the receiving node is received.

- The accompanying diagrams illustrate the format of an MSCP message that might be sent across the CI Bus.

Command Packets

Response Packet

DISK DATA TRANSFER COMMAND PACKET

31		0
COMMAND REFERENCE NUMBERS		
RESERVED	DRIVE UNIT NUMBER	
WRITE w/ READ MODIFIERS ETC.	RSVD	READ OPCODE WRITE
HOW MANY BYTES BYTE COUNT		
WHERE IN MEMORY OF VAX IS THIS BUFFER DESCRIPTOR FROM OR TO		
LOGICAL BLOCK NUMBER		

ETC.

END MESSAGE PACKET

31		0
COMMAND REFERENCE NUMBER		
RESERVED	UNIT NUMBER	
BAD BLOCK SUCCESS STATUS	BAD BL FLAGS *	ENDCODE
BYTE COUNT		
BUFFER DESCRIPTOR		

TAPE DATA TRANSFER COMMAND PACKET

31		0
COMMAND REFERENCE NUMBER		
RESERVED	UNIT NUMBER	
MODIFIERS	RSVD	OPCODE
BYTE COUNT		
BUFFER DESCRIPTOR		

Data Flow

The following is an abbreviated account of a data transfer between a node in a VAXcluster and an HSC disk server. The numbers correspond to the figures on pages 2-13 and 2-15. In this case, a "write" to the disk is done:

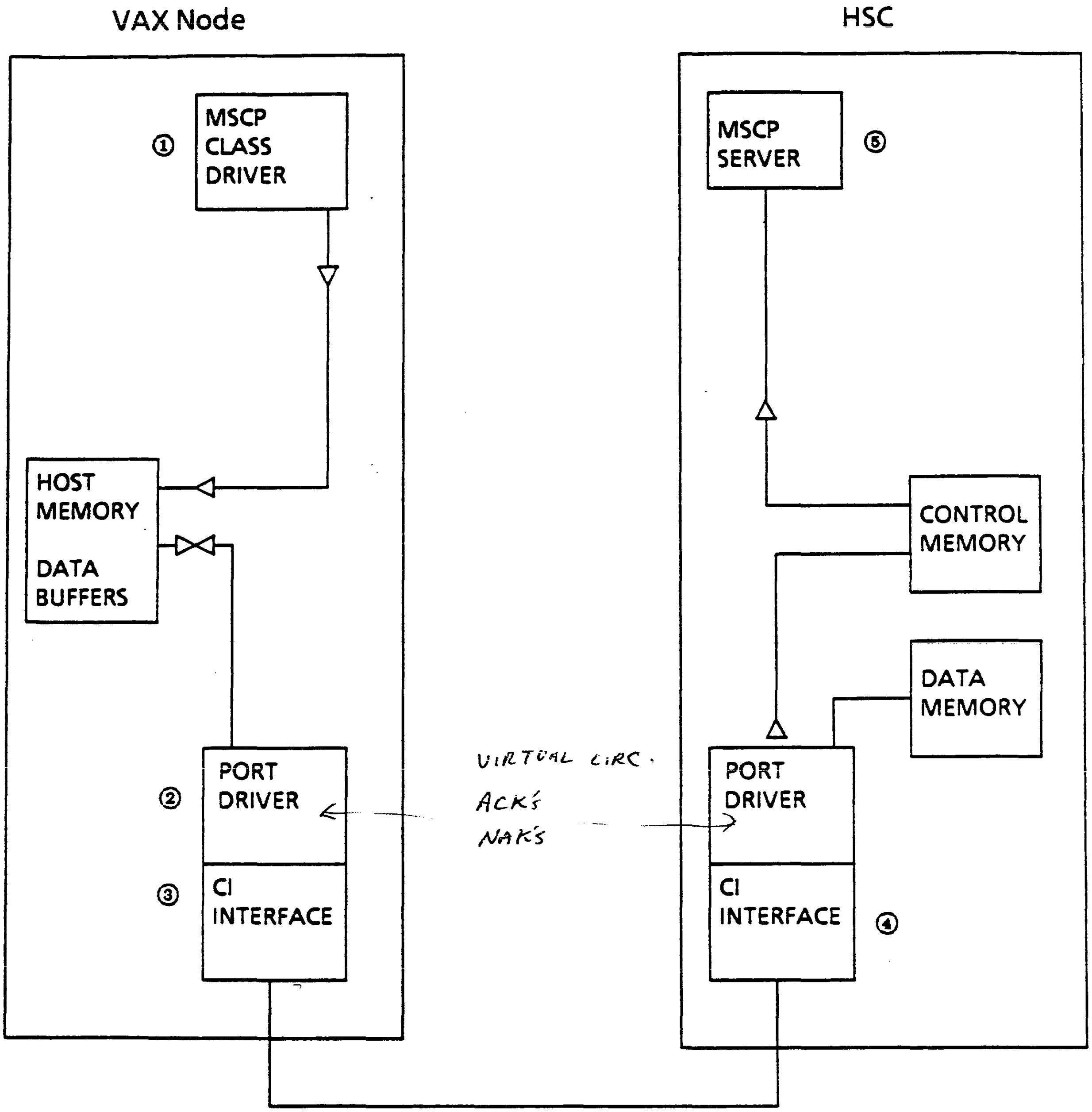
① The MSCP write command is issued by the host class driver.

② The CI Port driver adds an envelope HEADER, TRAILER to the MSCP command packet (packet type, packet length, destination, source).

③ The link board of the CI Port arbitrates for the CI Bus (listens for a quiet slot). The link board then adds sync and trailer bytes and transmits packet.

④ The destination link board (in HSC) issues an ACK.

⑤ In HSC, the MSCP message is stripped of the envelope by the port driver and handed to the class server.



Typical Data Transfer (Diagram 1)

Data Flow (Cont.)

⑥ The class server in HSC allocates data structures in data memory to handle the requested transfer and puts together an end message packet (but does not send it yet).

⑦ The CI Port of HSC issues series of send data requests, each request ending with an ACK received from host node.

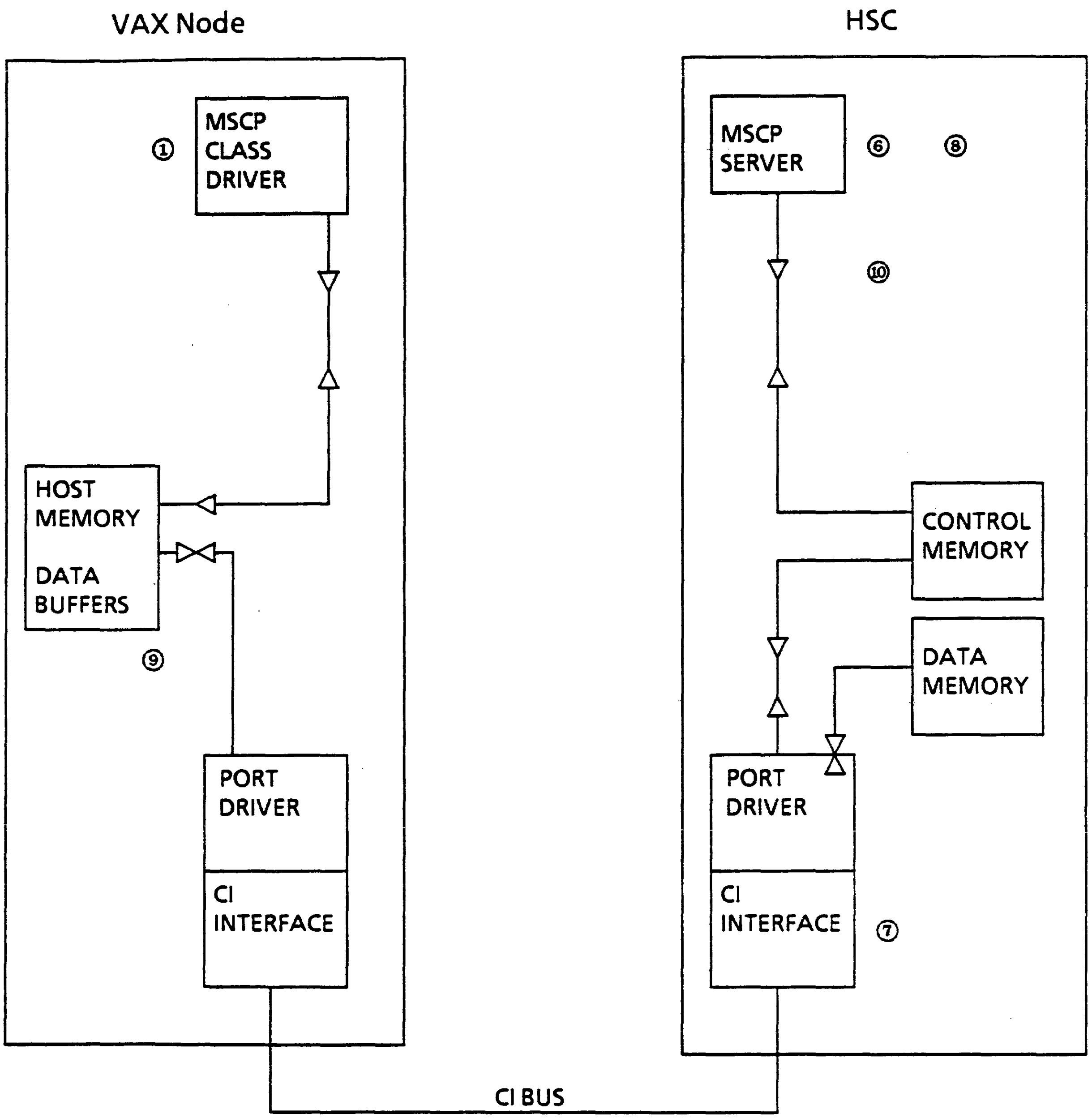
⑧ HSC software issues an SDI seek command to appropriate drive.

⑨ The host responds to these requests by sending the data packet and data, and then waiting for an ACK from the HSC CI Port.

The sequence continues until the transfer is complete. The last packet from the host contains a bit indicating the end of the data.

⑩ The HSC port server now sends an end message (previously put together by port server) back to the host node, satisfying the requirement that every MSCP message be completed with a response message.

NOTE: A transmission on the CI Bus is never considered "complete" until an ACK packet has been received from the correct node.



Typical Data Transfer (Diagram 2)

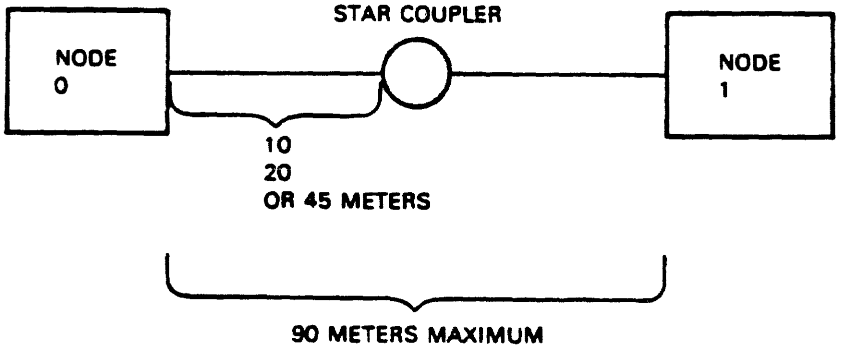
The Star Coupler

CAN BE 24 OR 32

- Provides central connection point for up to sixteen nodes of a CI Cluster.
- Transformer coupled connections -- CI cables can be added or removed without affecting cluster communications.
- Completely passive device -- no active electrical components, no power requirements.
- Two types of Star Couplers
  - SC008-AC for dual path, 8-node cluster
  - SC008-AD for dual path, 16-node cluster JUST PANEL

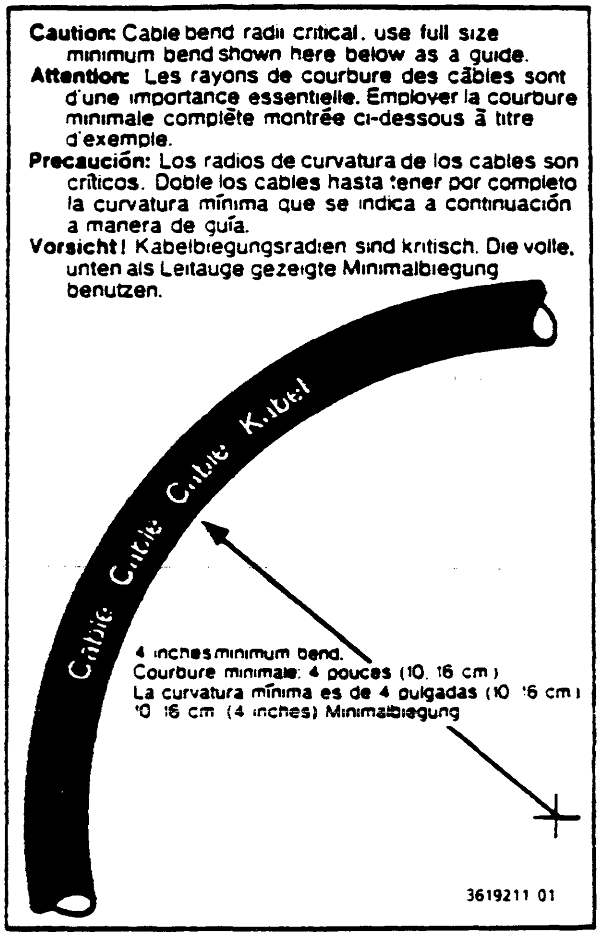
ORDER BOTH FOR COMPLETE 16 PORT DUAL

- Cable types
  - BNCI-XX = One pair (single path, two cables) LENGTH
  - BNCIA-XX = Two pairs (dual path, four cables)
- Maximum distance between nodes:



TK-10488

Bend Radius of CI Cables



Caution: Cable bend radii critical. use full size minimum bend shown here below as a guide.

Attention: Les rayons de courbure des câbles sont d'une importance essentielle. Employer la courbure minimale complète montrée ci-dessous à titre d'exemple.

Precaución: Los radios de curvatura de los cables son críticos. Doble los cables hasta tener por completo la curvatura mínima que se indica a continuación a manera de guía.

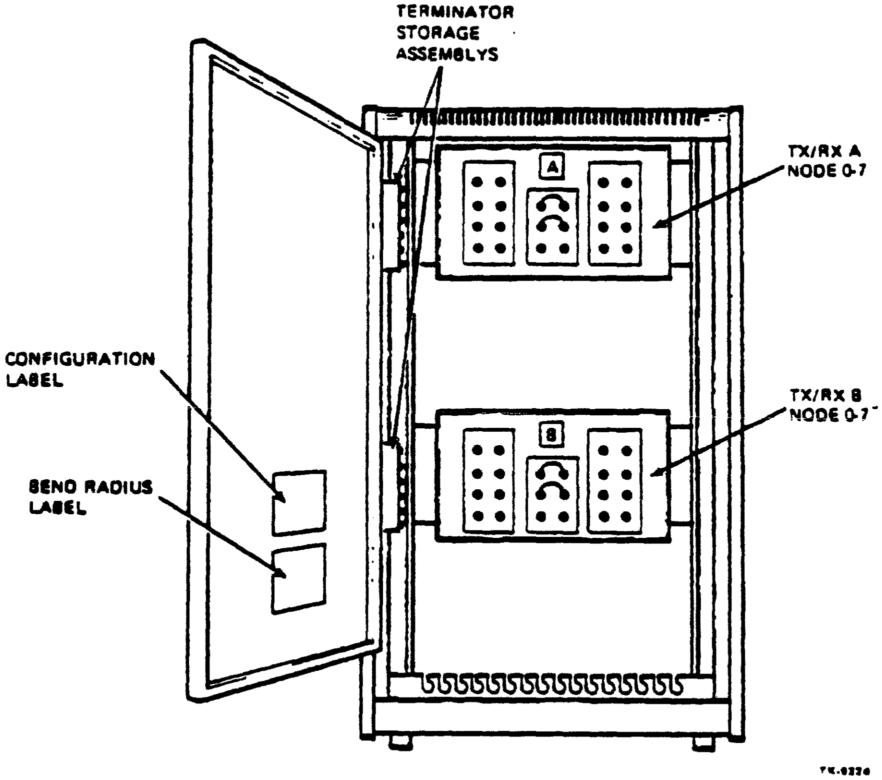
Vorsicht! Kabelbiegungsradien sind kritisch. Die volle. unten als Leitauge gezeigte Minimalbiegung benutzen.

4 inches minimum bend.
Courbure minimale: 4 pouces (10. 16 cm)
La curvatura mínima es de 4 pulgadas (10 16 cm)
10 16 cm (4 inches) Minimalbiegung

3619211 01

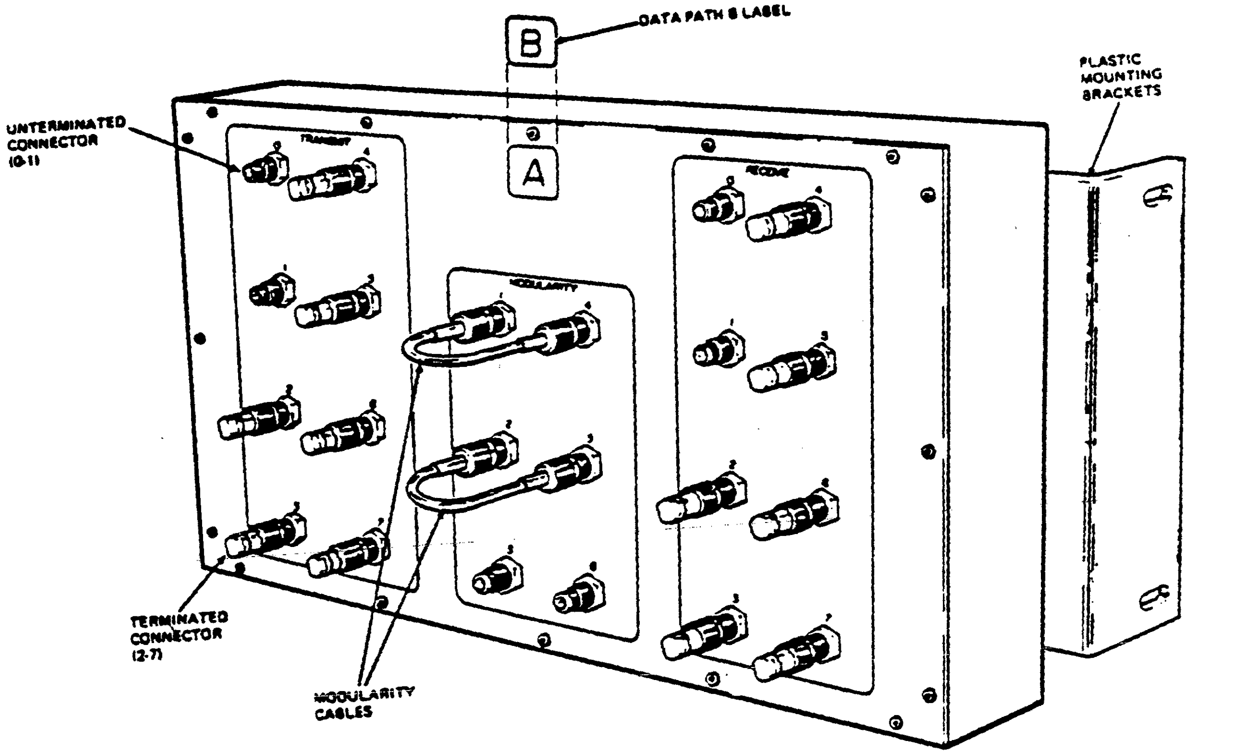
TK-9217

The Star Coupler Cabinet



The Star Coupler Box

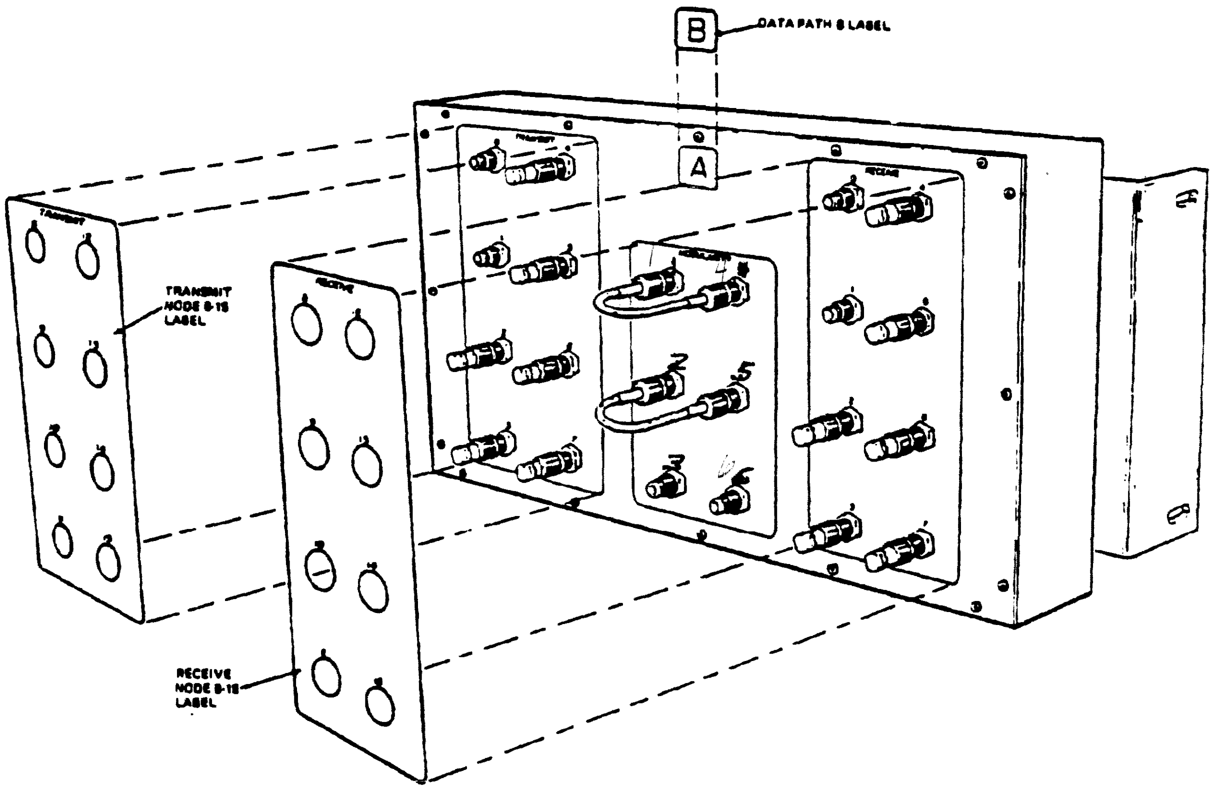
- Each box (panel) contains eight receive connections and eight transmit connections:


DATA PATH B LABEL
B
A
PLASTIC MOUNTING BRACKETS
UNTERMINATED CONNECTOR (0-1)
TRANSMIT
MODULARITY
RECEIVE
TERMINATED CONNECTOR (2-7)
MODULARITY CABLES

TK-9226

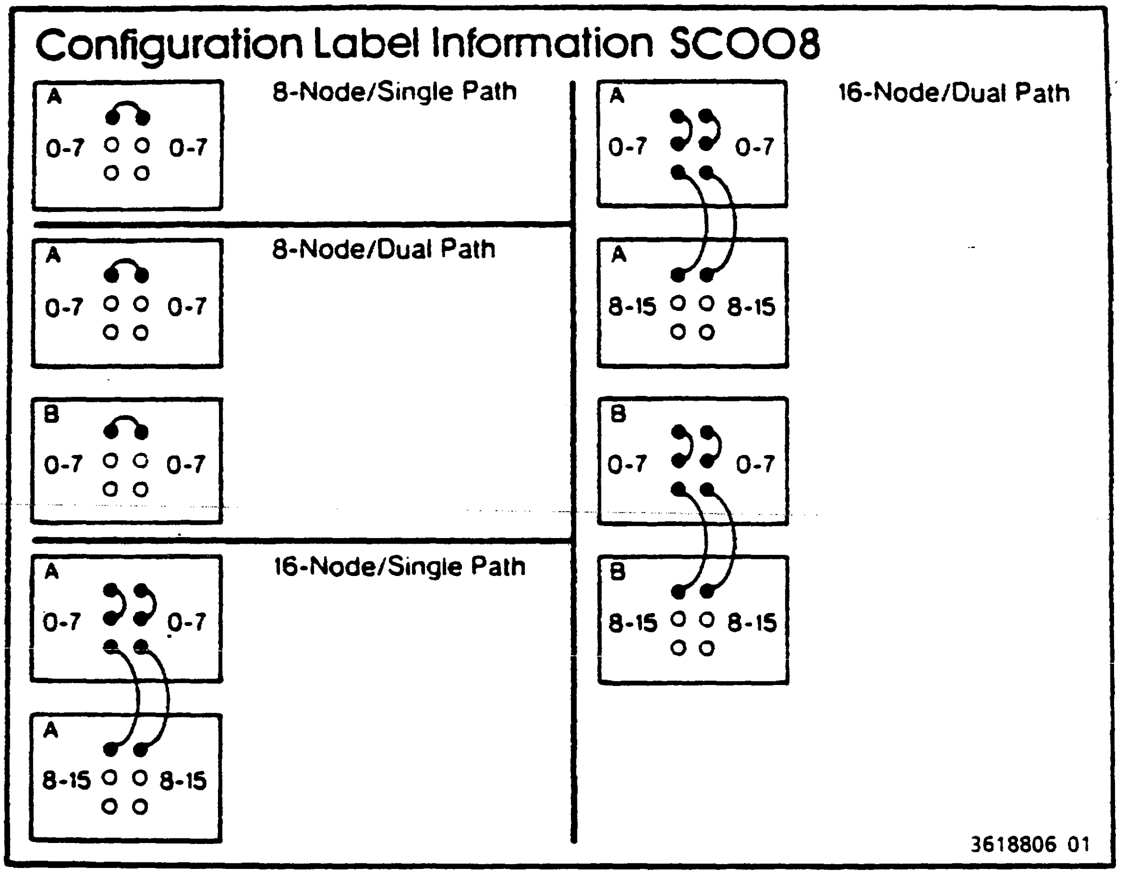
The Star Coupler Box (Cont.)

- When using a 16-node configuration, special labels are supplied for overlays:



Star Coupler Configuration

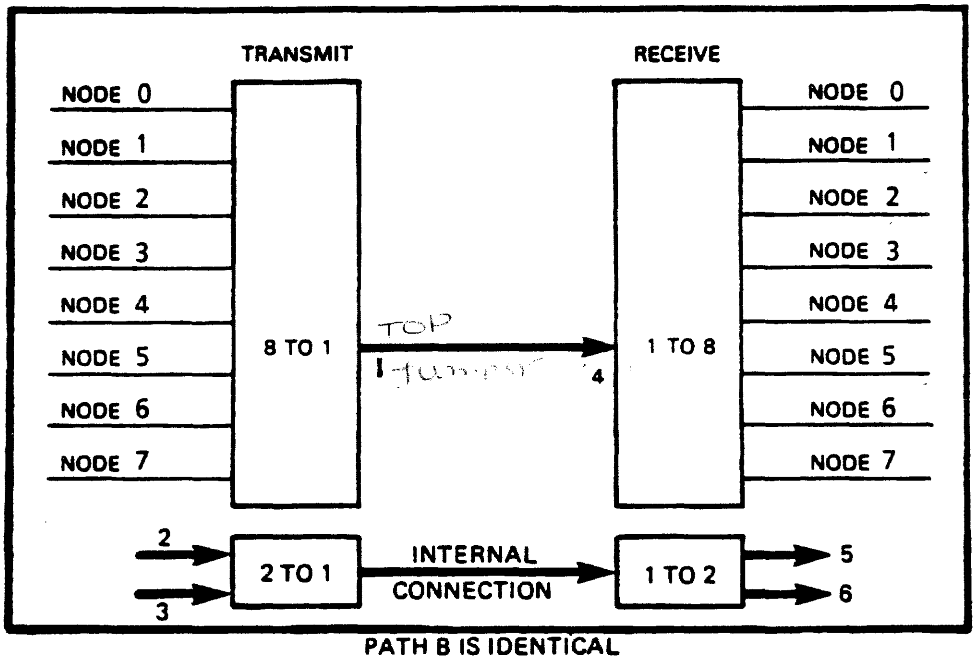
- Modularity cables determine how the Star Coupler is going to be used.
- Although available, single path configurations are not supported by Digital.



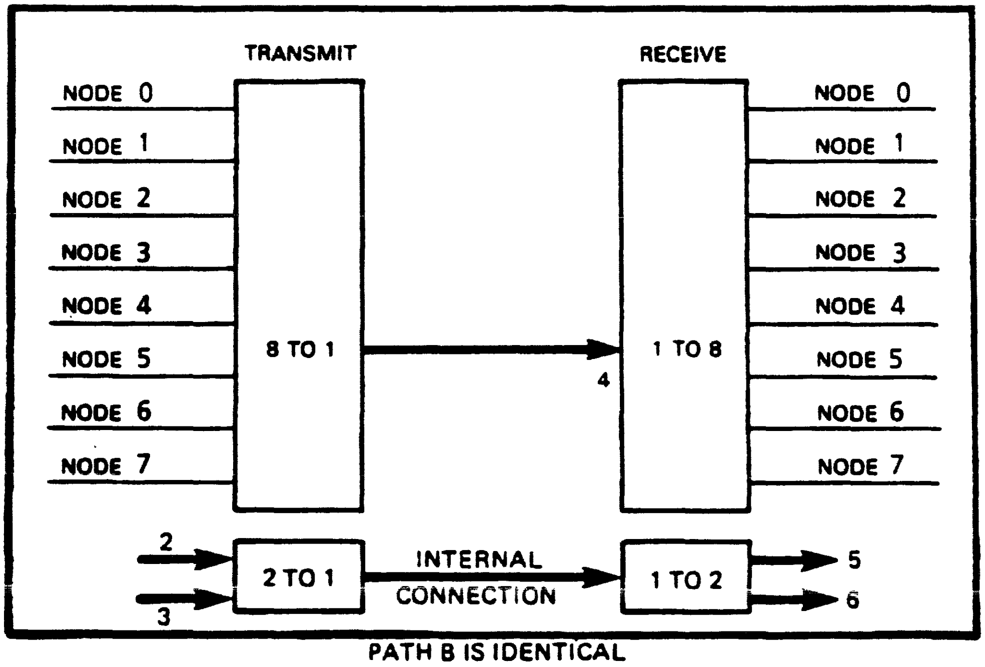
TK-8219

Star Coupler Configuration (Cont.)

- An 8-node, dual path configuration electrically looks like this:



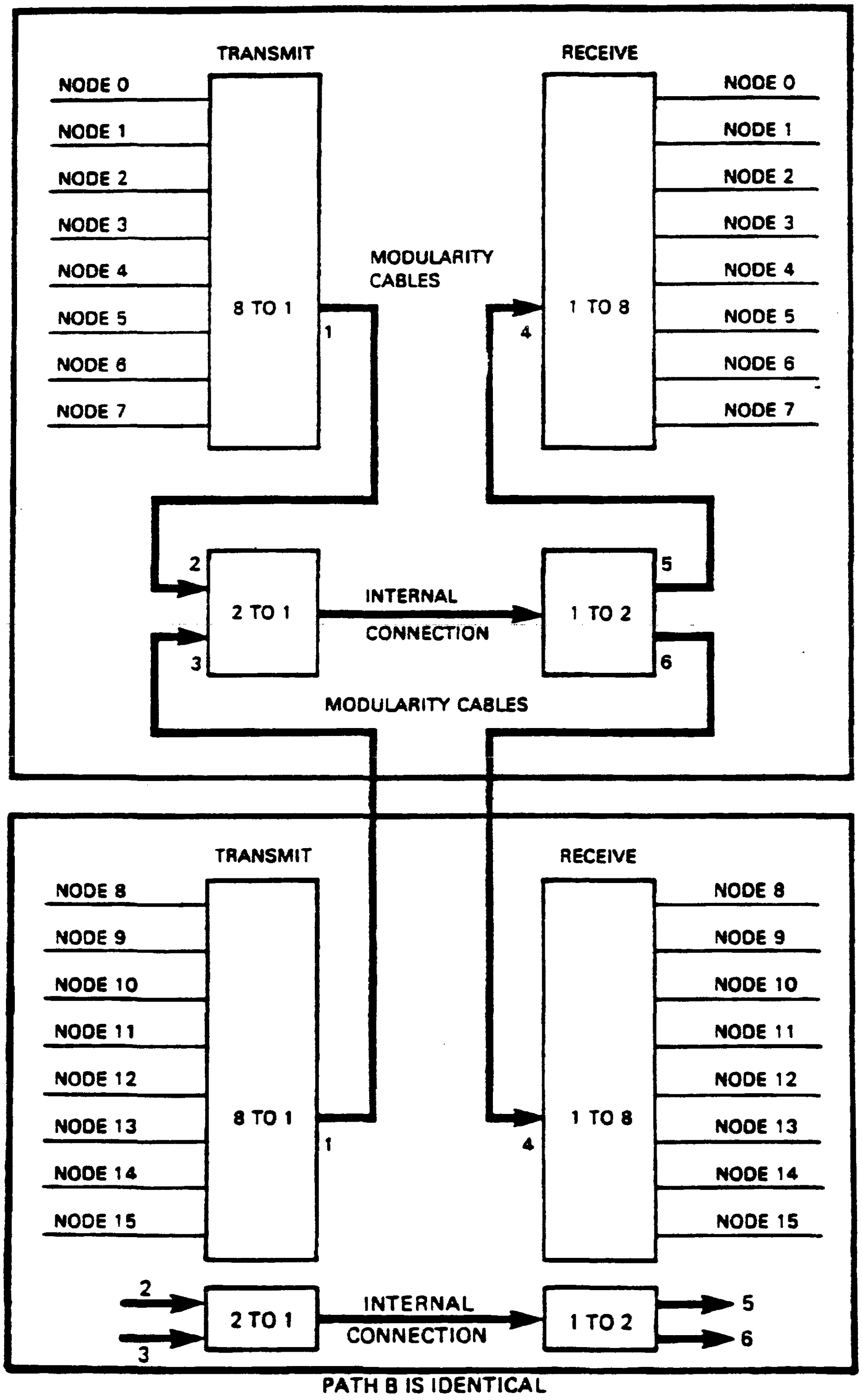
8-Node Configuration (Path A)



8-Node Configuration (Path A)

Star Coupler Configuration (Cont.)

- A 16-node, dual path configuration electrically looks like this (only one path shown):



TK-10508

CISCE (CI Star Coupler Extender)

- Used to extend a CI cluster from 16 nodes to 24 nodes.

- Maximum number of VAX processors allowed on CI cluster remains at 16: additional nodes beyond 16 must be HSC controllers.

- VAXcluster prerequisites for CISCE use:

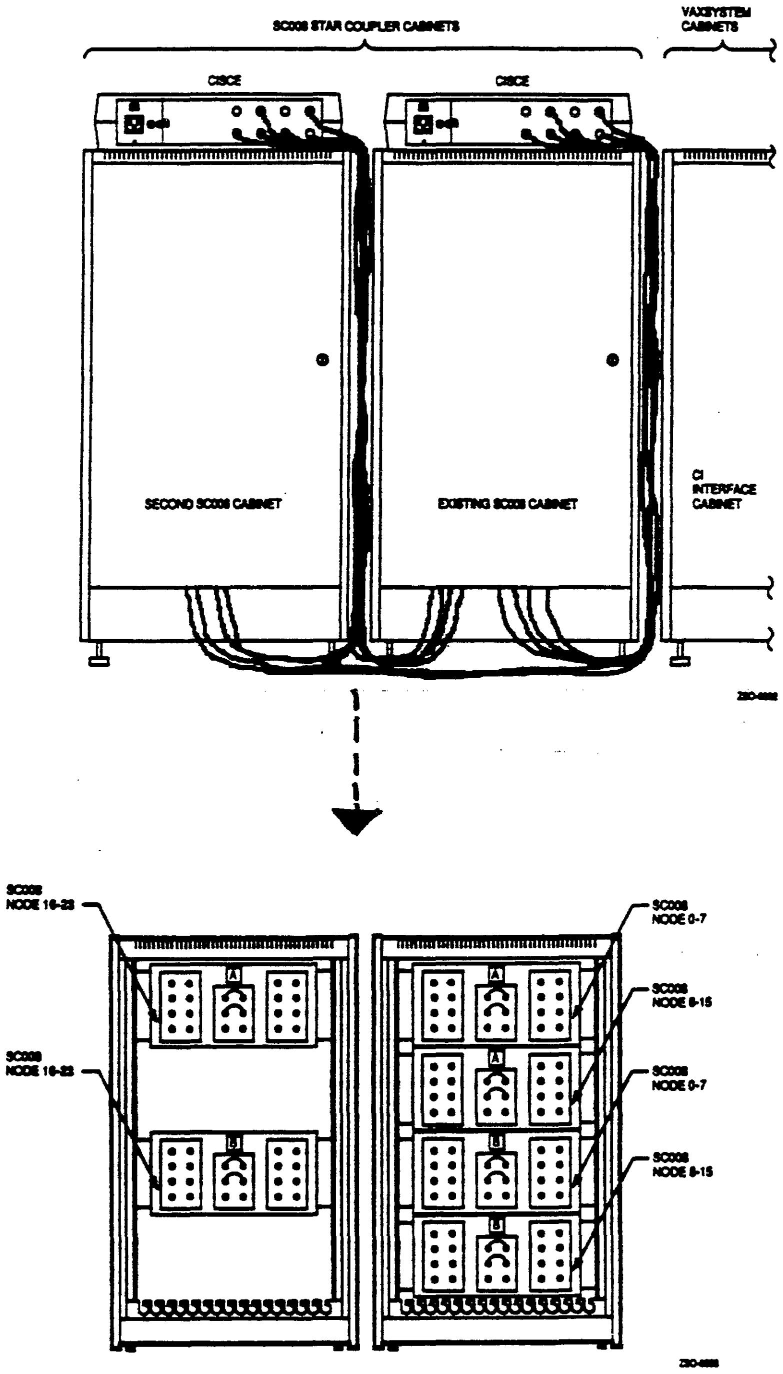
  a. All nodes must use L0118 Link module.
  b. All nodes must meet minimum hardware revision levels as specified in System Revision Control Document.
  c. VAXcluster should be running VMS 4.7 or later.
  d. HSC nodes should be running CRONIC 3.7 or later.

- Requires the following additional hardware:

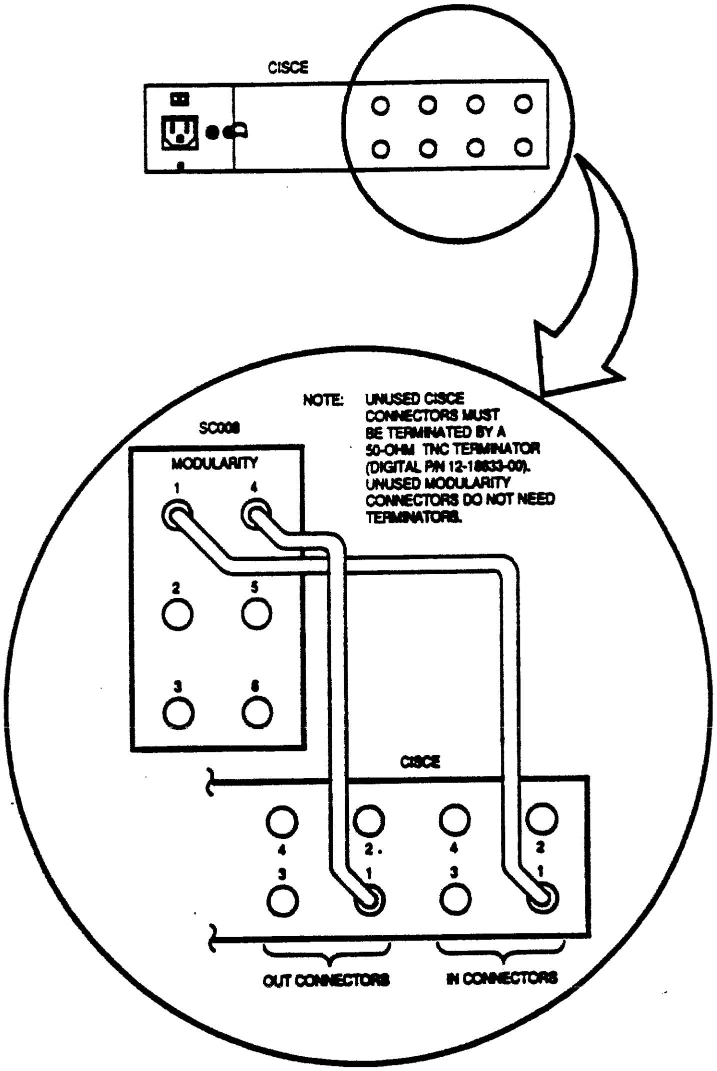
  a. A second star couple cabiniet containing two SC008 panels.
  b. Two CISCE amplifier boxes (one for each path).
  c. Four meter CISCE cable to connect amplifier box to Star coupler cabinet.
  d. Separate power receptacle for amp boxes

CISCE (CI Star Coupler Extender)

- The following diagrams illustrate the external layout of the CISCE.

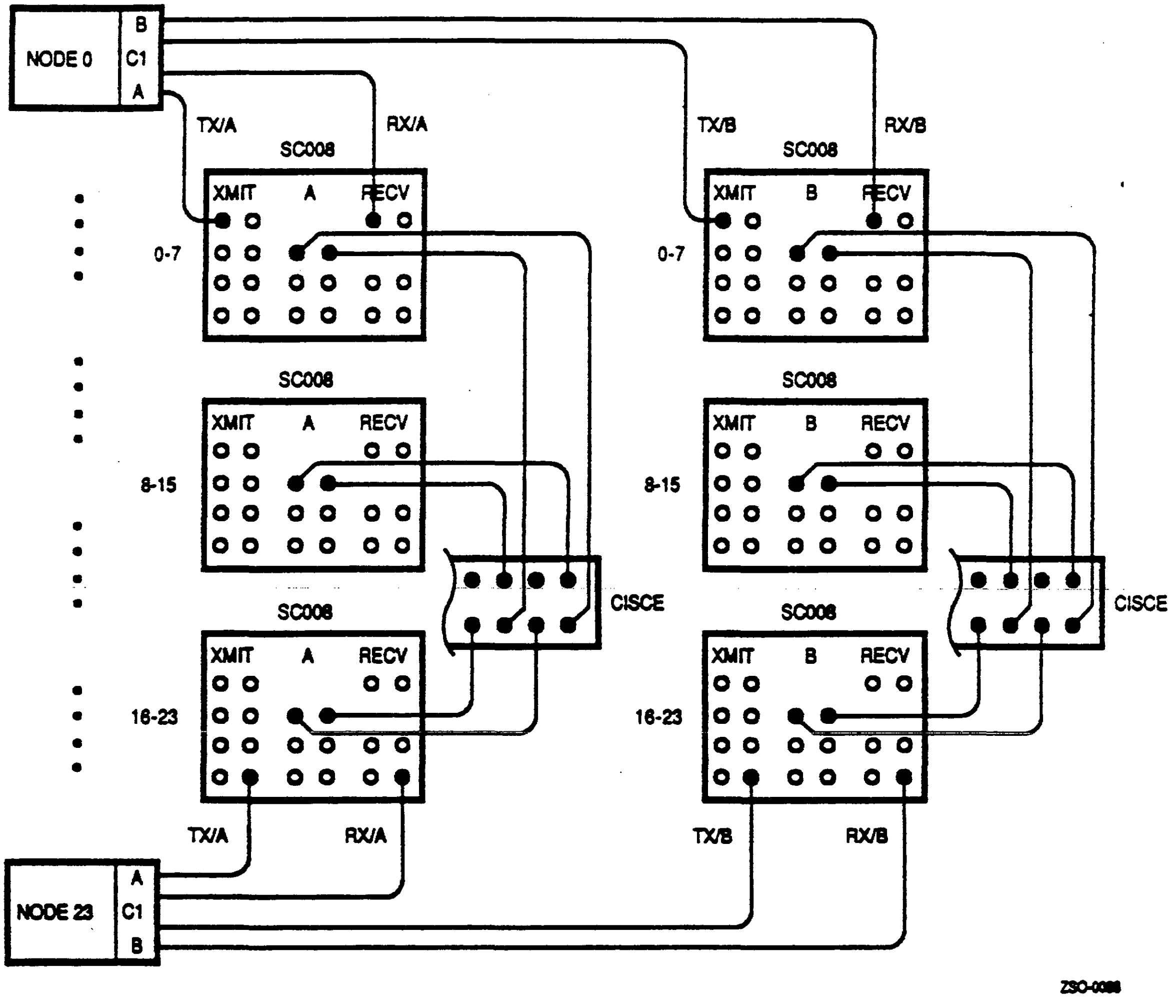


- o CISCE amplifier boxes are housed in a DECserver 200 common package.
- o Each box contains a power supply and a four amplifiers on one module.
- o Each amplifier shares four common signal sources (IN connectors).
- o Modularity output (connector 1) from SC008 panel is routed to one of the four IN CONNECTORS of the CISCE. Properly amplified output is sent through OUT CONNECTORS of the CISCE to modularity input (connector 4) of each available SC008 panel.



NOTE: UNUSED CISCE CONNECTORS MUST BE TERMINATED BY A 50-OHM TNC TERMINATOR (DIGITAL P/N 12-18833-00). UNUSED MODULARITY CONNECTORS DO NOT NEED TERMINATORS.

o The following diagram illustrates a 24-node dual path configuration.



THE CI INTERFACE

The CI Interface

Lesson Introduction

The CI Interface in each VAX is an intelligent interface between the System Bus of that VAX and the CI Bus. The microprocessor in the CI Interface runs its own microcode and can function fairly independently of the VAX processor. We will discuss the physical description and the installation for each of the different types of CI Interfaces.

Lesson Objectives

1. Identify each of the different CI Interfaces.
2. Describe the difference between the CI750, CI780, CIBCA, and CIBCI.
3. Describe the physical characteristics of a CI Interface.
4. Install a CI Interface onto a CMI-, SBI-, or BI-based VAX.
5. Understand the distribution of power to the CI Interface.
6. Set the node address on a CI Port.

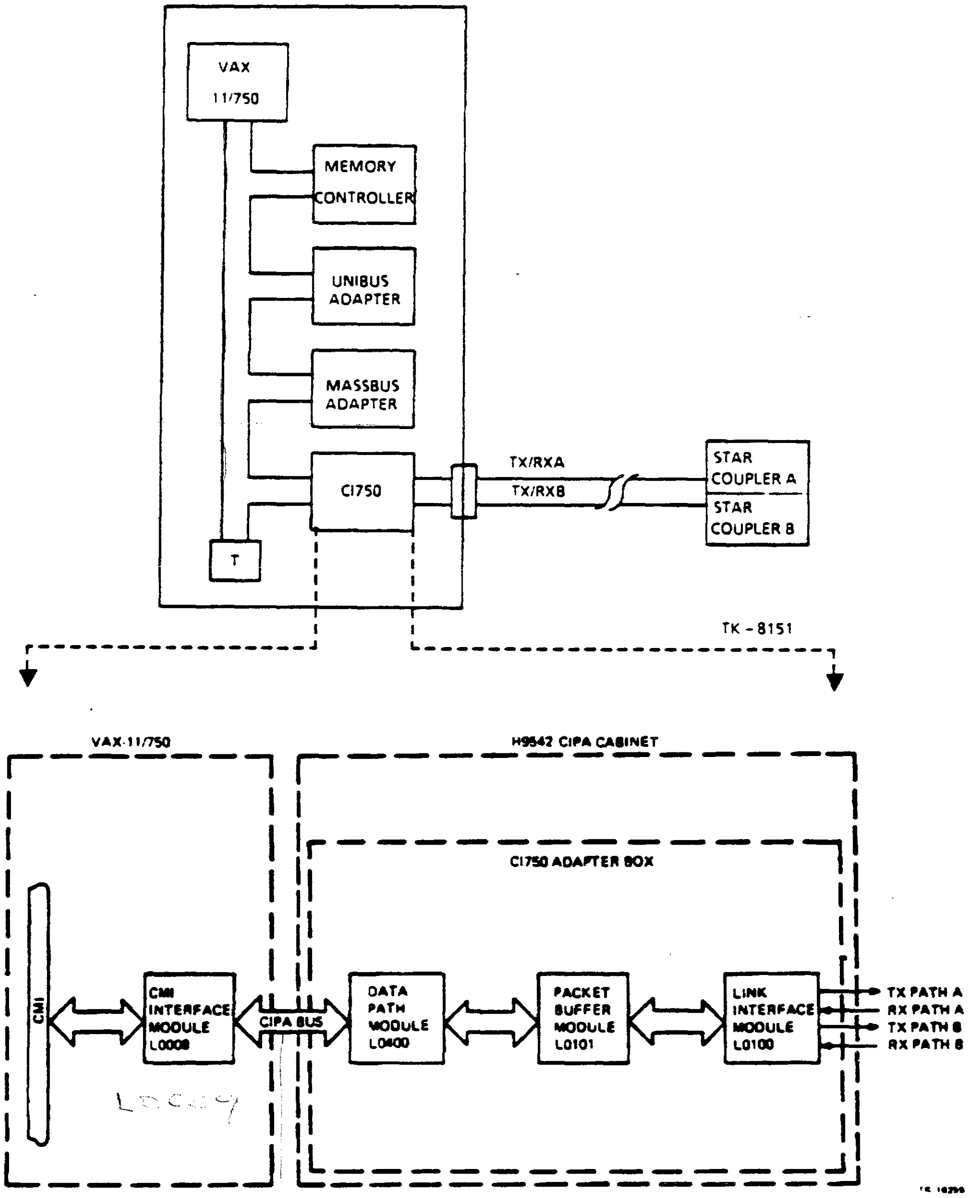
Lesson Outline

I. CI750
II. CI780
III. CI780 in a VAX 8600/8650
IV. CIBCI

CI750 INTERFACE

The CI750 Port

- Operates from the VAX 750 CMI bus.
- Functions as a buffered communications port adapter for the VAX 750 computer and other nodes within the CI Cluster.
- Reduces CPU overhead by performing data buffering, address translation, and serial encoding/decoding within the cluster environment.

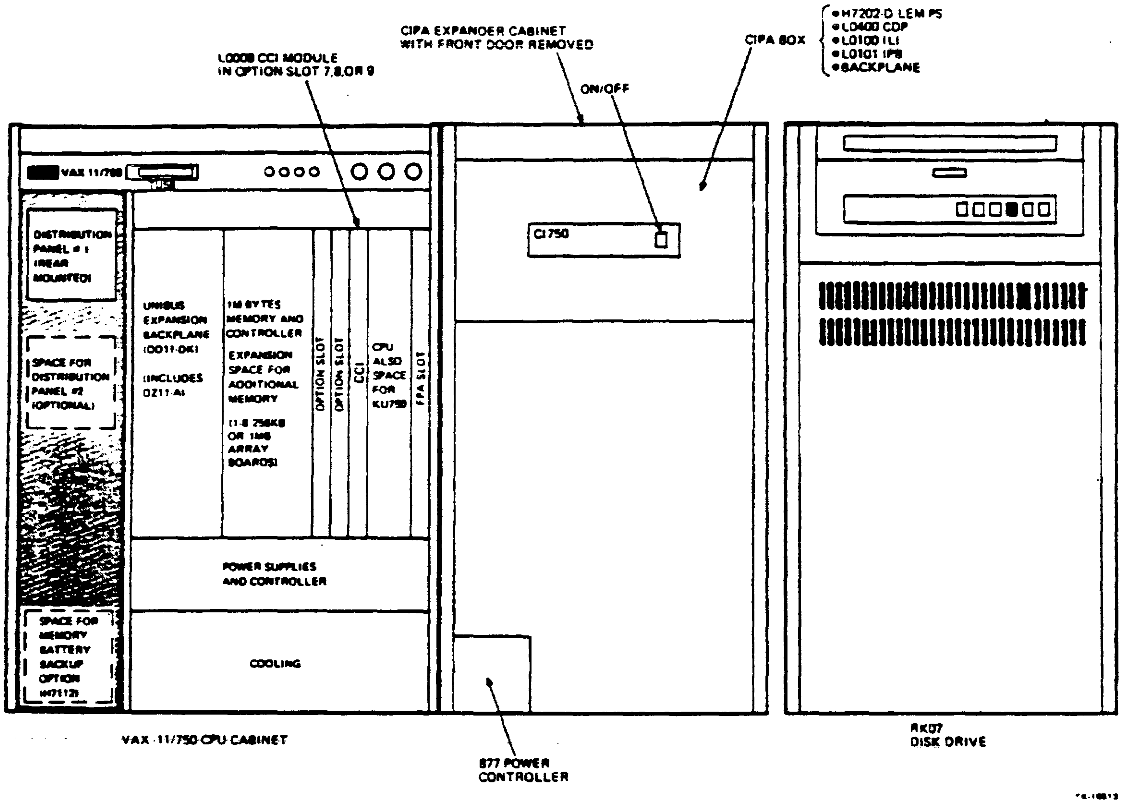


CIPA

Block Diagram of the CI750 Adapter

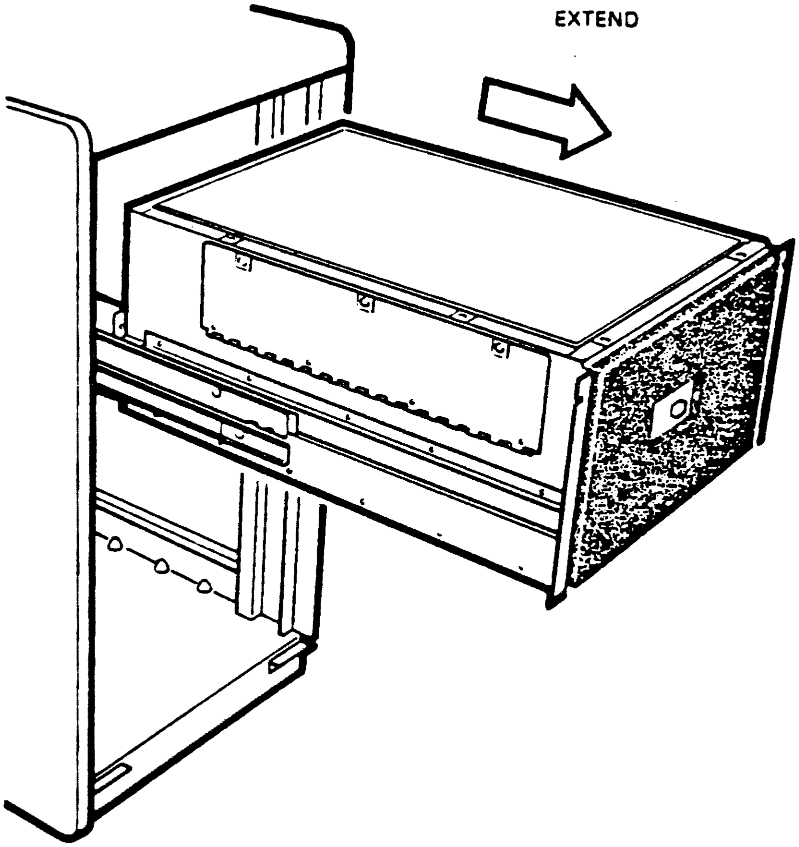
CI750 Installation Highlights

- The CMI Interface module (L0009) is normally placed in option slot 7 of the CPU cabinet.
- The CI750 base address is normally F3E000, which is determined by jumpers on the backplane slot of the L0009 module. Note that to obtain this address, all jumpers are OUT.
- A priority plug on the L0009 module provides the CI750 with the proper Bus Grant selection, therefore no CMI backplane jumpers are needed.
- The CMI Arbitration jumpers should be set such that the CI750 is the LOWEST priority device of all the options present.
- Normally, the Slow CMI jumper will be OUT.
- CI Bus arbitration and quiet slot times are determined by jumpers on the CIPA box backplane.
- Boot timer default is determined by jumpers on the CIPA box backplane.
- Node address switches are on the link board in the CIPA box.



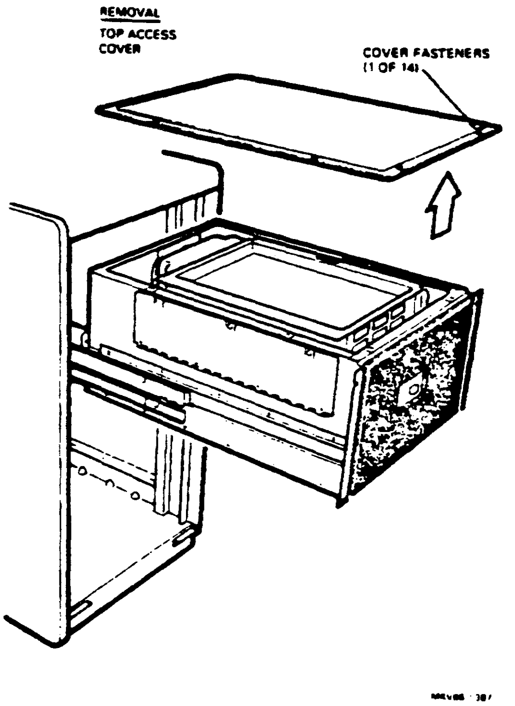
Front View of the CI750

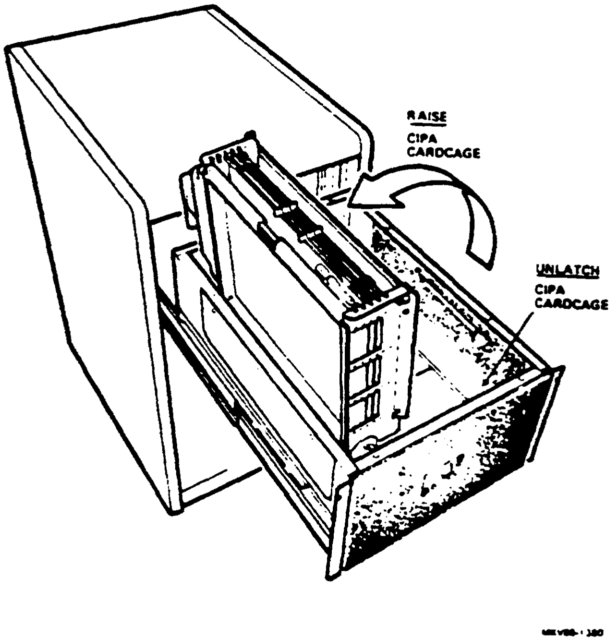
Extracting the CIPA Logic Box



MKV86-1360

CIPA Box Module Access





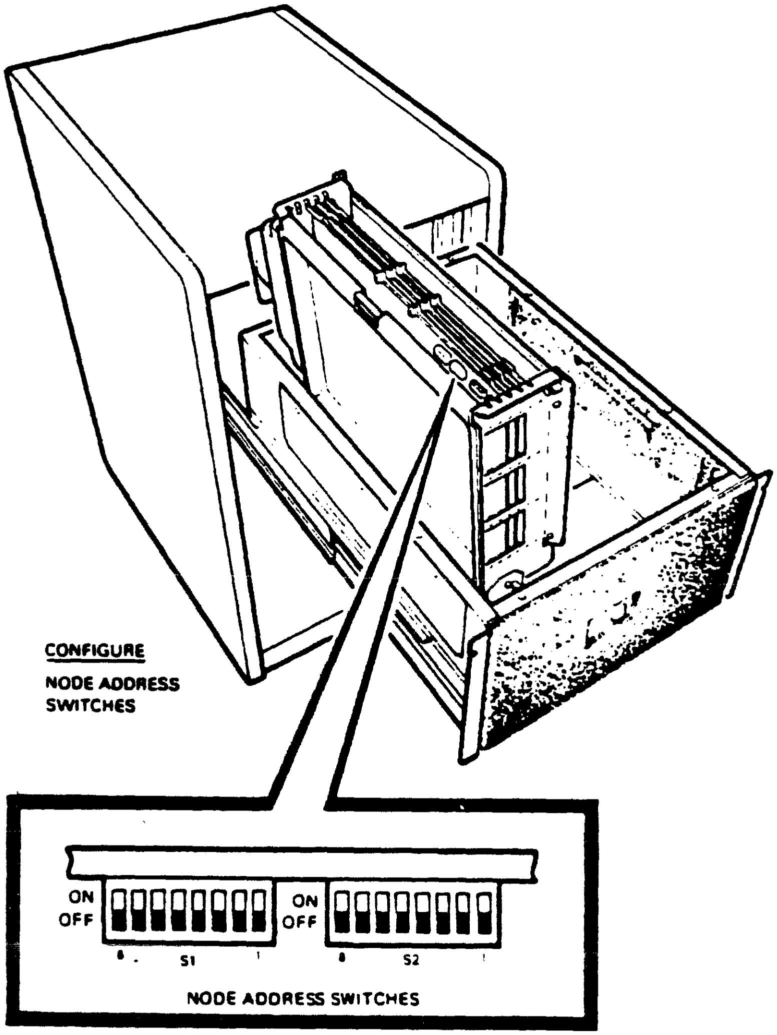
CIPA Box Module Access (Cont.)

1	MAINT – CDP (L0400) ALTERNATE
2	MAINT – IPB (L0101) ALTERNATE
3	ILI (L0100)
4	NORMAL – IPB (L0101)
5	NORMAL – CDP (L0400)

FOR CHIP CHASING ALLOWS BOARD TO BE PUT ON TOP OF STACK TO GET AT CHIPS

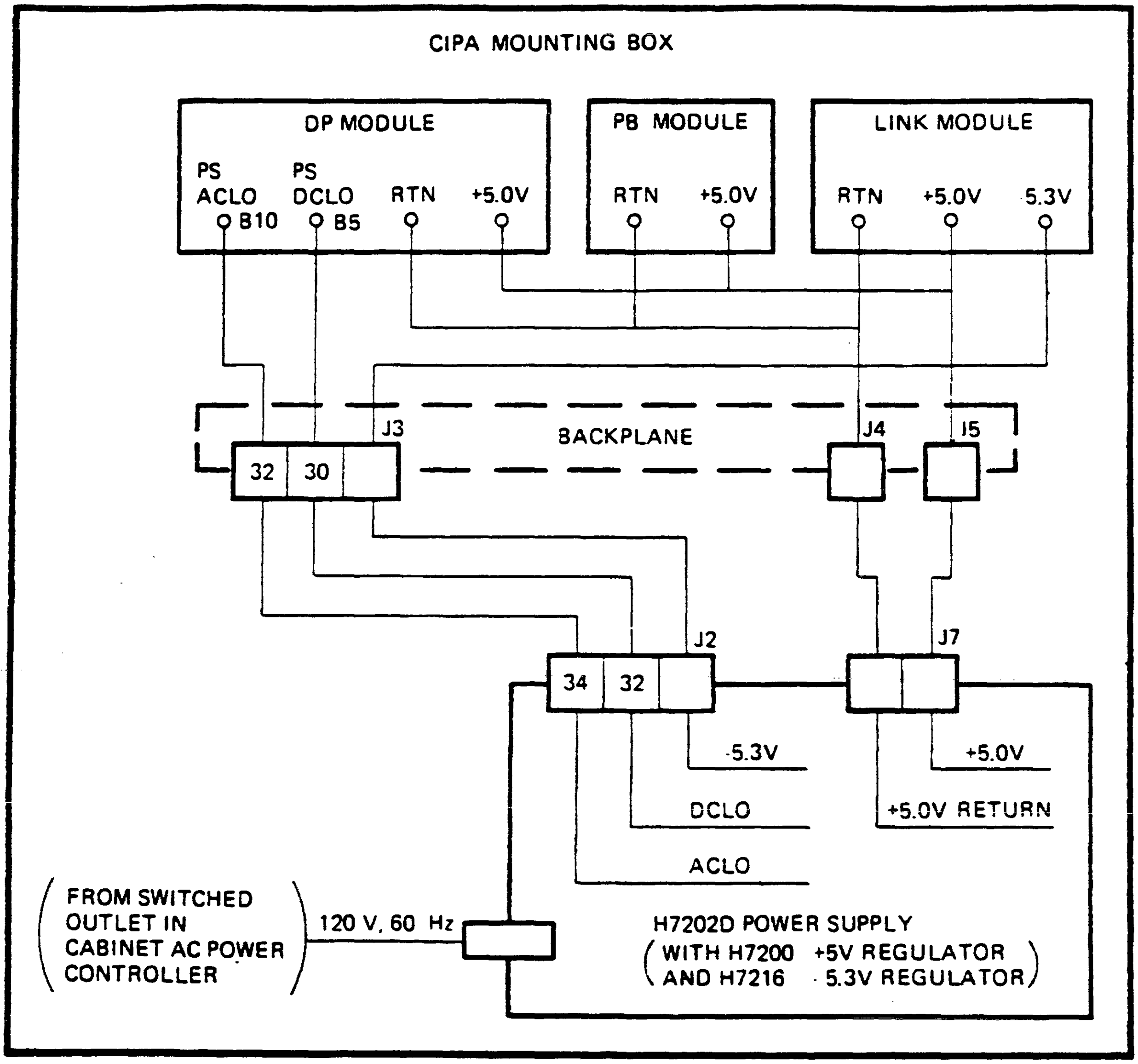
TK-10507

CI750 CIPA Module Utilization



MKV86-1363

CIPA Mounting Box -- Module Access



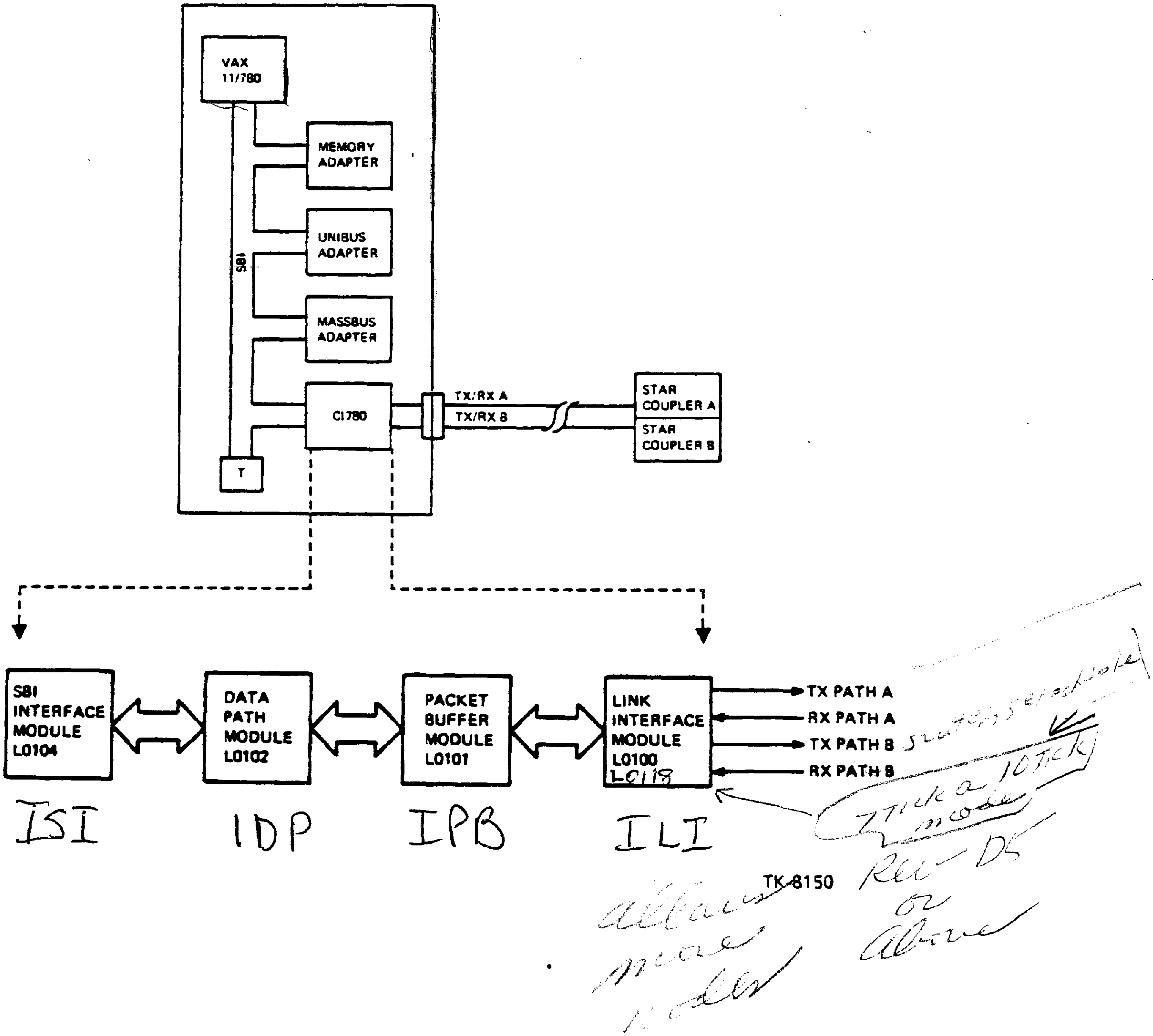
MKV85-0504

CIPA Power Distribution

CI780 INTERFACE

The CI780 Port

- Operates from the SBI bus on the VAX 780.
- The CI780 may be installed in any 4-inch option slot in either the standard CPU cabinet or an expansion cabinet of the host system.
- The CI780 option consists of four extended hex L-series modules and a pressed pin backplane.
- The diagrams below illustrate how the CI780 fits into the CPU configuration.



TK-8150

CI780 Installation Highlights

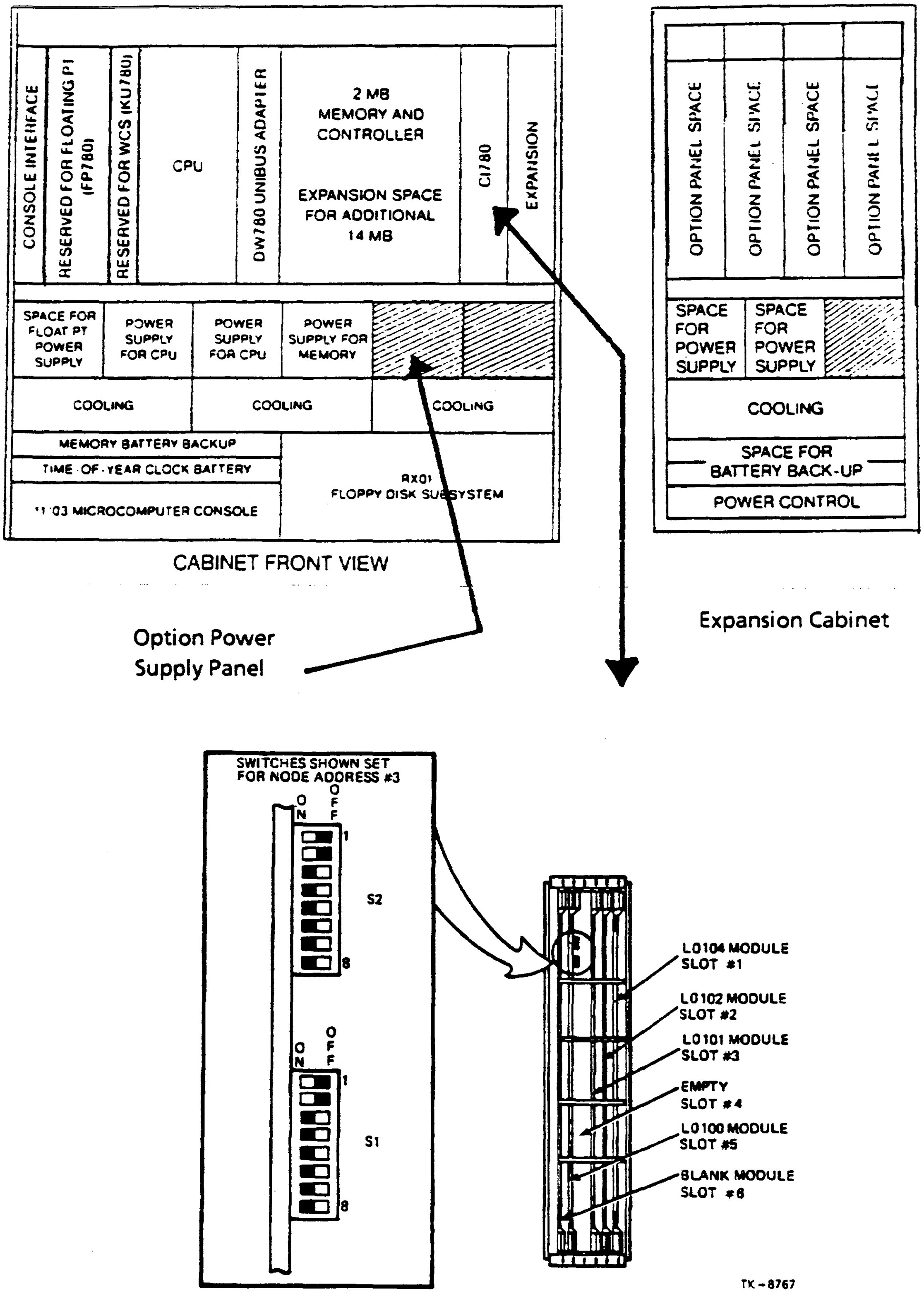
- The CI Interface can be placed in an option slot in either the main CPU cabinet or the expansion cabinet.

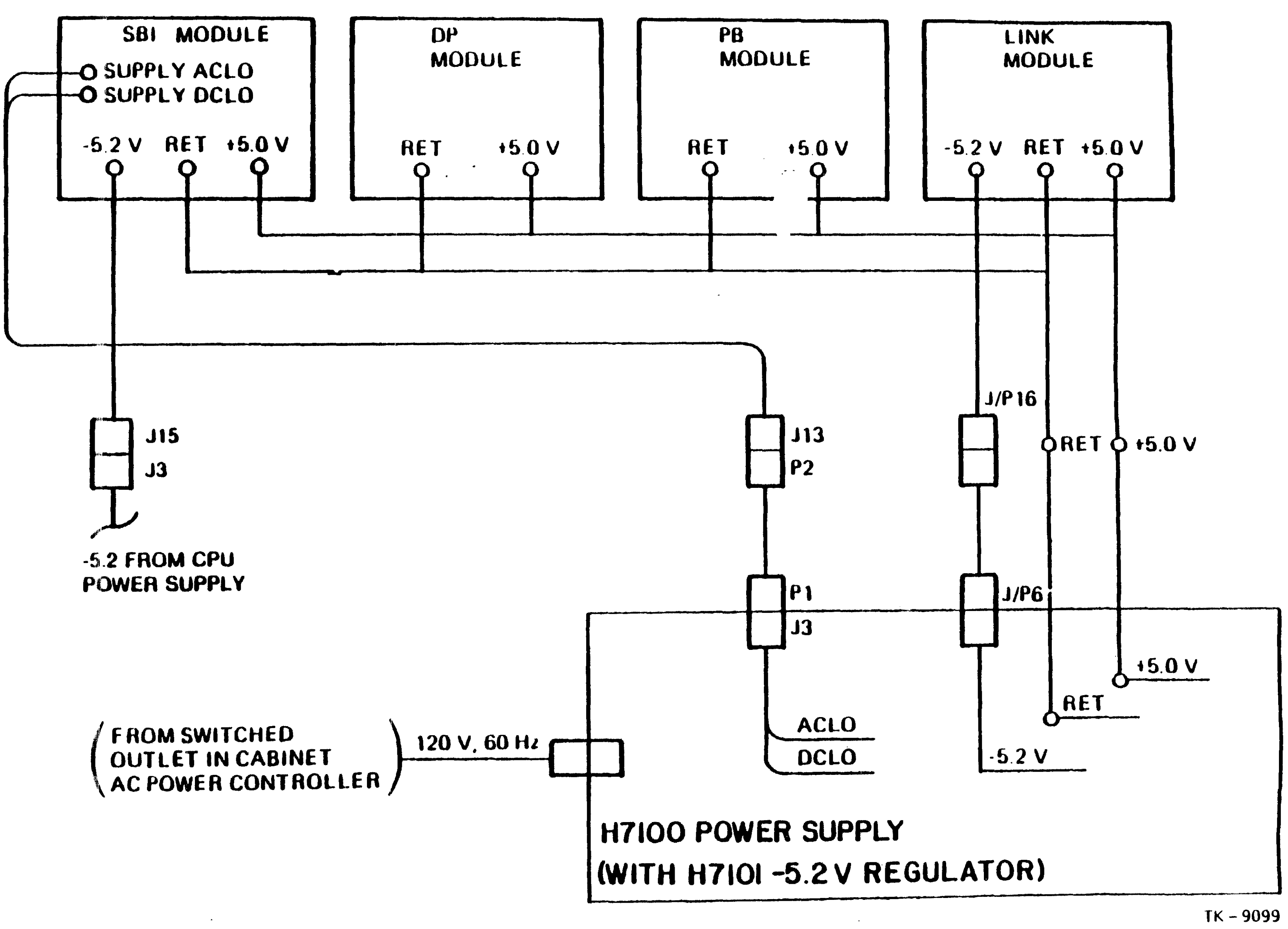
- Pay attention to power supply for the interface: -5.2V is needed (comes with the H7100).

- The CI Interface is normally shipped with the TR arbitration level already set to 14 (physical address 2001C000) and the SBI BR priority level set to 4.

- Boot timer, quiet slot time, and arbitration times are determined by jumpers on the backplane of the CI780 assembly.

The CI Port in the VAX 11-780



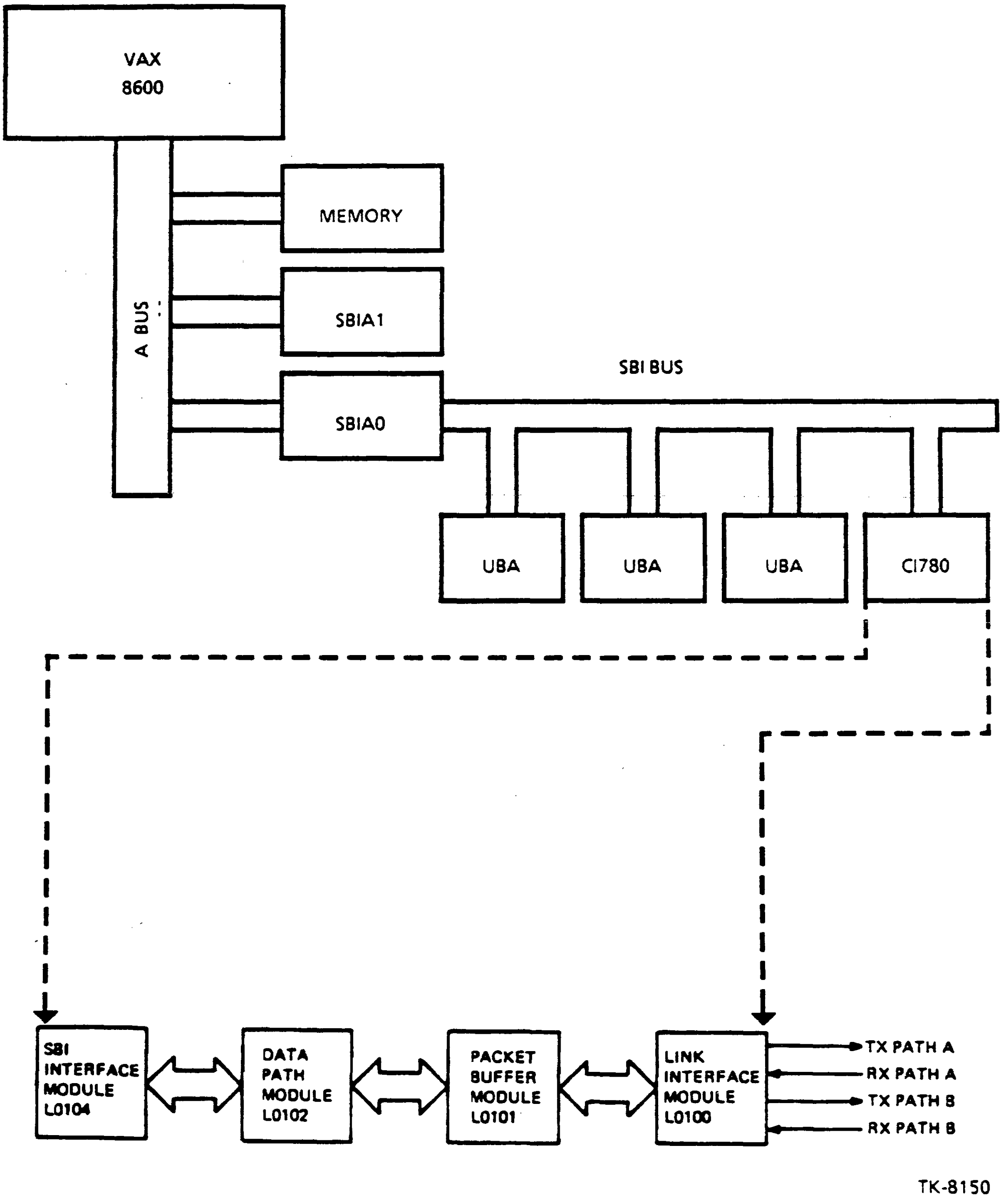


CI780 Power Distribution

VAX 8600/8650

The CI780 on the VAX 8600/8650

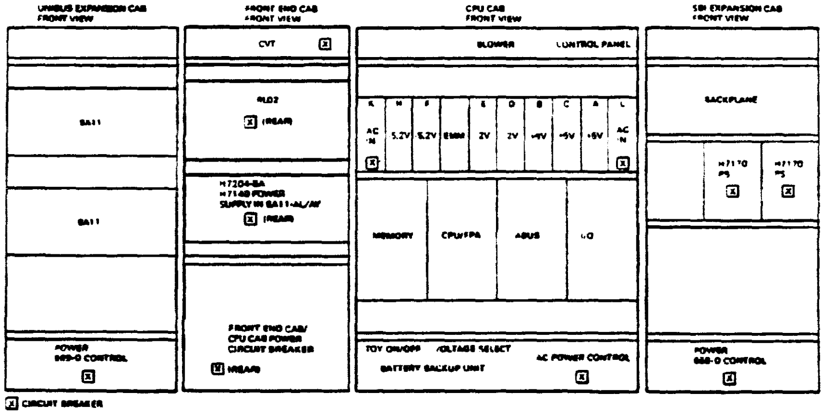
The CI Port hardware for the VAX 86xx CPU family resides on the SBI Bus. Therefore, the same four boards that compose the CI780 for the VAX 780 are used on the VAX 8600 and 8650.



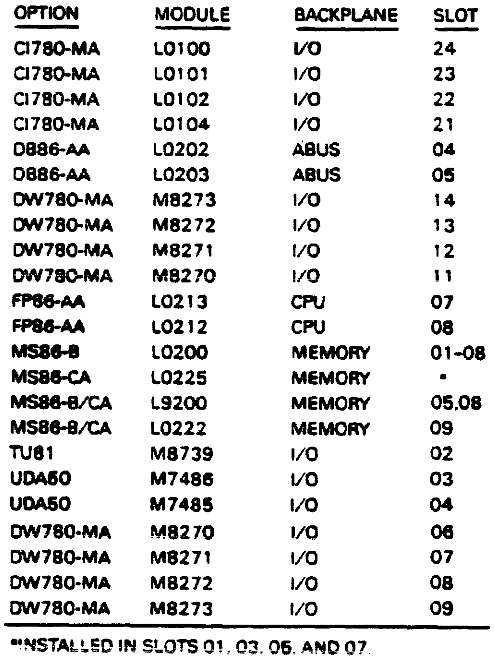
Installing the CI780 on the VAX 8600/8650

- Installing the CI780 adapter on the VAX 86xx is easier than installing one on the VAX 780 because you do not have to worry about power supplies or backplane installation. In the 86xx, the necessary power supplies and backplane are already there.

- The CI780 boards go in slots 21 through 24 on the I/O cardcage as shown below.



MR - 14665

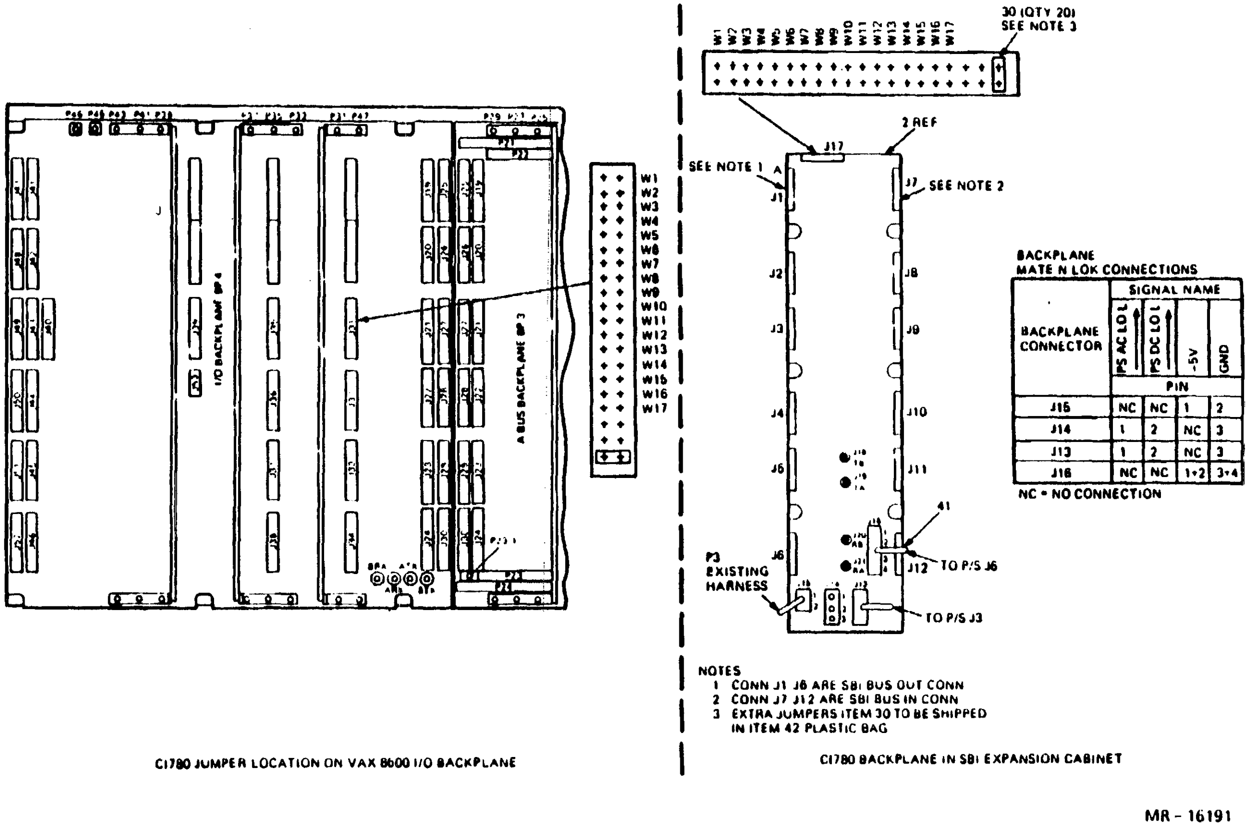


OPTION	MODULE	BACKPLANE	SLOT
CI780-MA	L0100	I/O	24
CI780-MA	L0101	I/O	23
CI780-MA	L0102	I/O	22
CI780-MA	L0104	I/O	21
DB86-AA	L0202	ABUS	04
DB86-AA	L0203	ABUS	05
DW780-MA	M8273	I/O	14
DW780-MA	M8272	I/O	13
DW780-MA	M8271	I/O	12
DW780-MA	M8270	I/O	11
FP86-AA	L0213	CPU	07
FP86-AA	L0212	CPU	08
MS86-B	L0200	MEMORY	01-08
MS86-CA	L0225	MEMORY	*
MS86-B/CA	L9200	MEMORY	05,08
MS86-B/CA	L0222	MEMORY	09
TU81	M8739	I/O	02
UDA50	M7486	I/O	03
UDA50	M7485	I/O	04
DW780-MA	M8270	I/O	06
DW780-MA	M8271	I/O	07
DW780-MA	M8272	I/O	08
DW780-MA	M8273	I/O	09

*INSTALLED IN SLOTS 01, 03, 05, AND 07.

MR-0188-0113

Internal Option Module Installation



BACKPLANE
MATE N LOK CONNECTIONS

BACKPLANE CONNECTOR	SIGNAL NAME			
	PS AC LO L	PS DC LO L	-5V	GND
	PIN			
J15	NC	NC	1	2
J14	1	2	NC	3
J13	1	2	NC	3
J16	NC	NC	1+2	3+4

NC = NO CONNECTION

NOTES
1 CONN J1 J6 ARE SBI BUS OUT CONN
2 CONN J7 J12 ARE SBI BUS IN CONN
3 EXTRA JUMPERS ITEM 30 TO BE SHIPPED
IN ITEM 42 PLASTIC BAG

CI780 Backplane Jumper Locations

CIBCI VAXBI INTERCONNECT

OBSOLETE

The CIBCI Interface

- The CIBCI is the interface used to connect a VAXBI system (82xx, 83xx, 85xx, 87xx, 88xx) to the CI Cluster.
- The CIBCI adapter is partitioned into two separate hardware interfaces: one host processor interface and one computer interconnect port adapter interface. These interfaces consist of the following major components.
  - Two Eurocard T-series modules

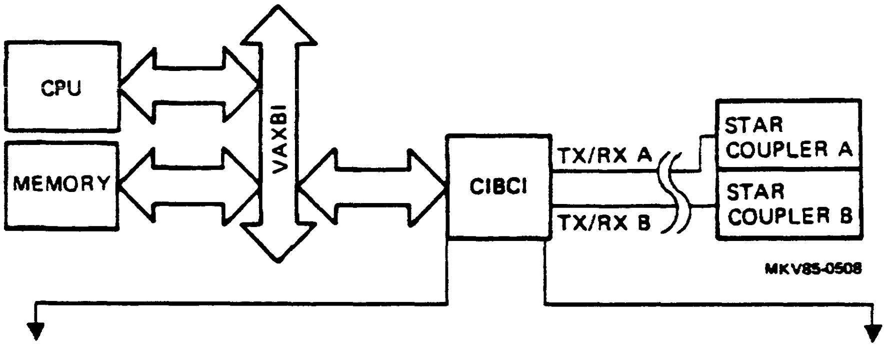
    a. Adapter control module (BAC) T1017

    b. Adapter data module (BAD) T1018
  - Three extended hex L-series modules

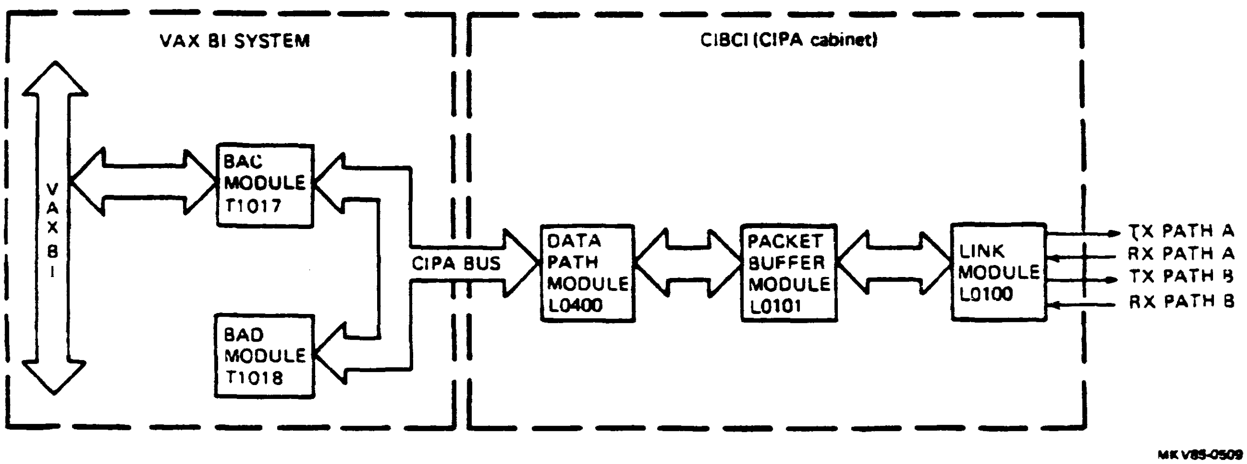
    a. Link interface module (ILI) L0100

    b. Packet buffer module (IPB) L0101

    c. Data path module (CDP) L0400



MKV85-0508

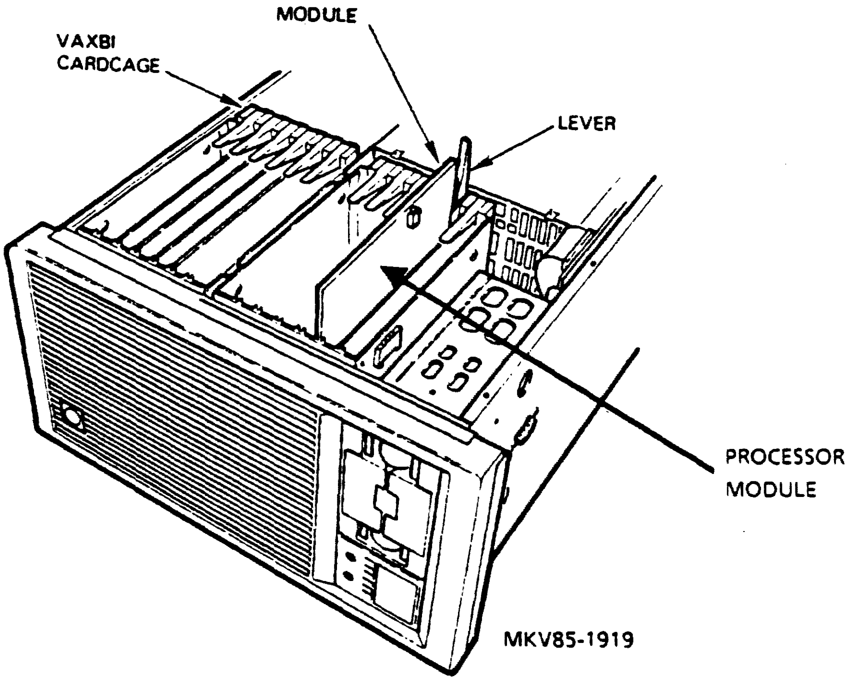


MKV85-0509

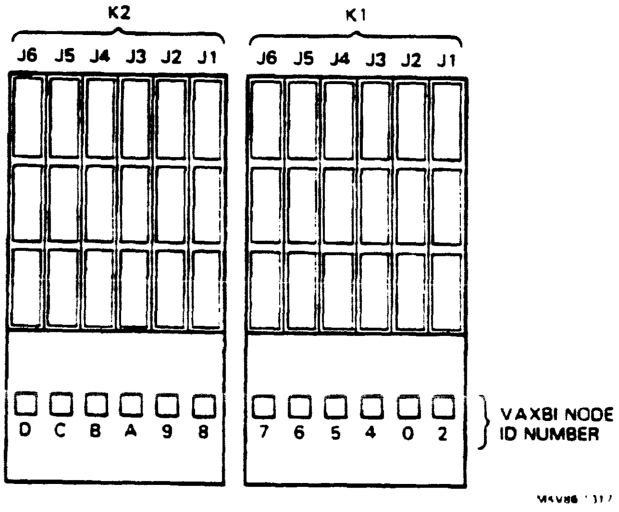
Location of the CIBCI Adapter (VAX 82xx, 83xx)

- The following figures show the module layout for VAX 82xx and 83xx family computers. The CIBCI host processor interface goes in any two unoccupied option slots.

- **Slot K1J1 always contains the primary CPU module. The CIBCI option is installed so that the BAC module is always to the left of the BAD module.**



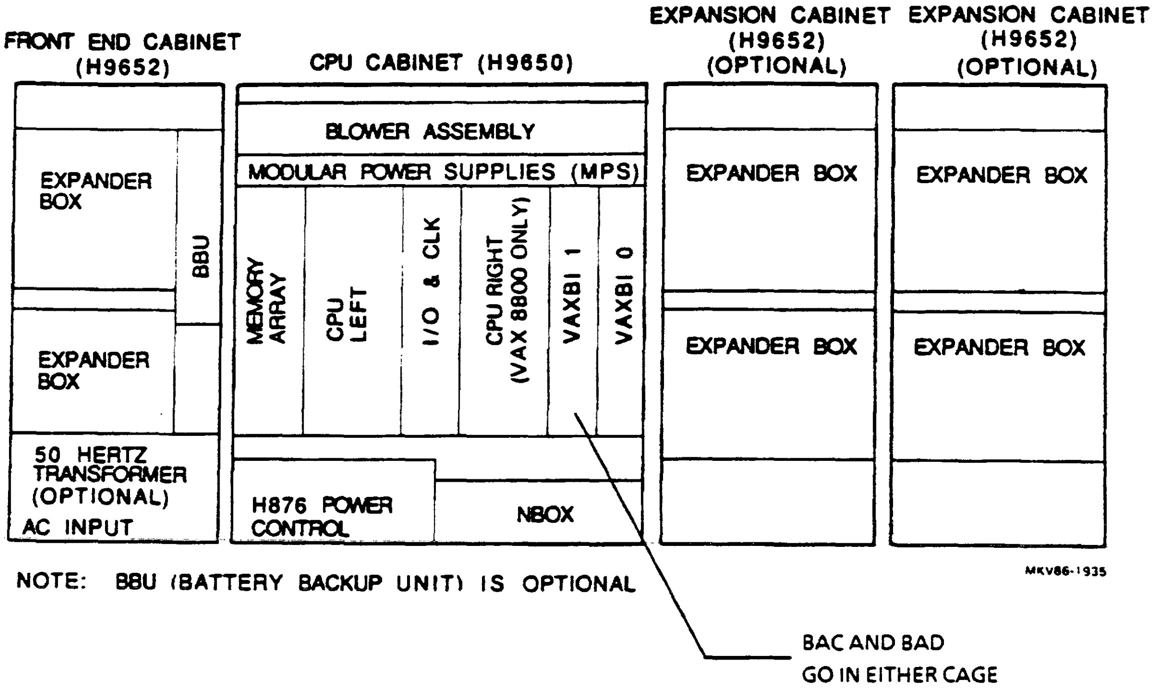
The 12-VAXBI Slot Cabinet



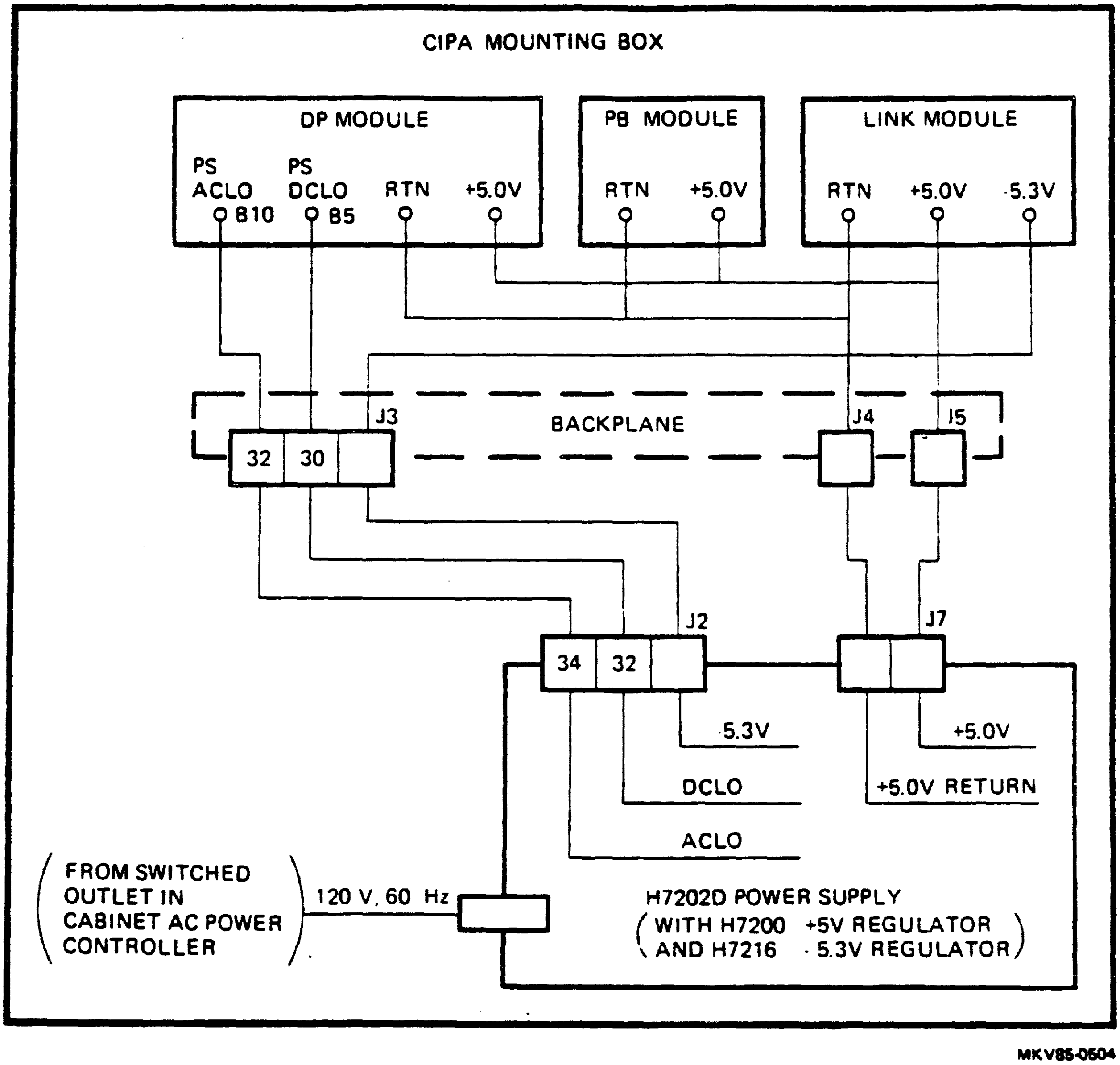
Top View of VAXBI Slot Cabinet

Location of the CIBCI Adapter (VAX 85xx, 87xx, 88xx)

The CIBCI host processor interface (BAC and BAD modules) goes in either BI cage assembly, while the remaining boards of the port adapter interface go in the external CIPA cabinet.


FRONT END CABINET (H9652)
CPU CABINET (H9650)
EXPANSION CABINET (H9652) (OPTIONAL)
EXPANSION CABINET (H9652) (OPTIONAL)
EXPANDER BOX
BBU
EXPANDER BOX
50 HERTZ TRANSFORMER (OPTIONAL)
AC INPUT
BLOWER ASSEMBLY
MODULAR POWER SUPPLIES (MPS)
MEMORY ARRAY
CPU LEFT
I/O & CLK
CPU RIGHT (VAX 8800 ONLY)
VAXBI 1
VAXBI 0
H876 POWER CONTROL
NBOX
EXPANDER BOX
EXPANDER BOX
EXPANDER BOX
EXPANDER BOX
NOTE: BBU (BATTERY BACKUP UNIT) IS OPTIONAL
MKV86-1935
BAC AND BAD GO IN EITHER CAGE

VAX 8700/8800 System Cabinet Layout



CIPA Power Distribution

CIBCA VAXBI INTERCONNECT

The CIBCA-AA Interface

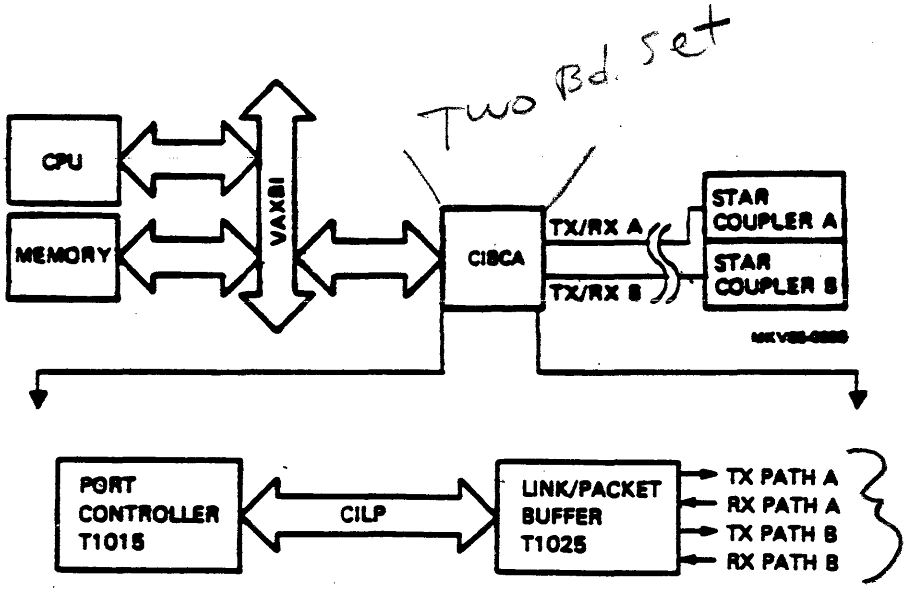
- The CIBCA-AA is used to connect a VAXBI system (83xx,85xx,88xx, 62xx) to the CI cluster.

- The CIBCA-AA adapter is composed of two separate hardware modules:

  a. T1015 port controller
  b. T1025 link/packet buffer

- The CIBCA is functionally identical to the CIBCI: provides buffered parallel-to-serial communications between the BI bus and the CI bus.

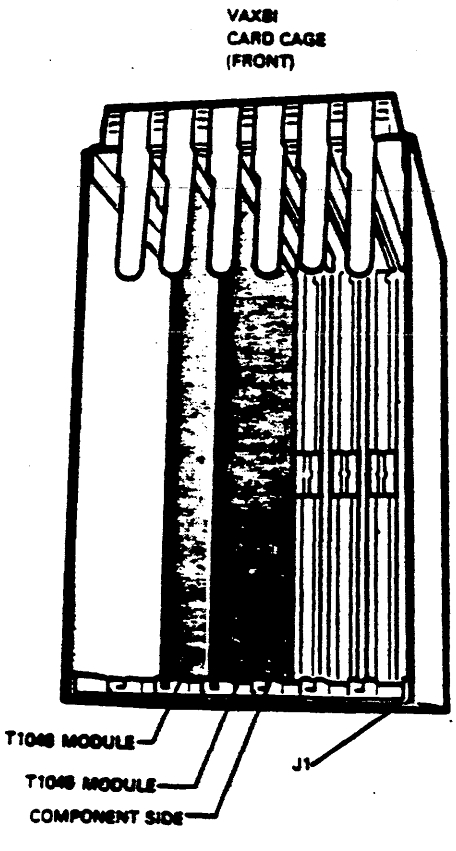
- The T1015 (port controller) manages the operation of the T1025 via the CI Link Protocol (CILP) bus.

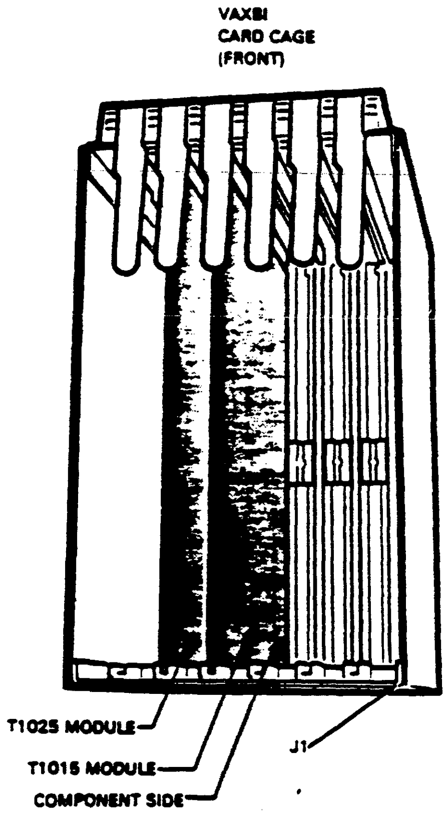


Two Bd. set

CIBCA Installation Highlights

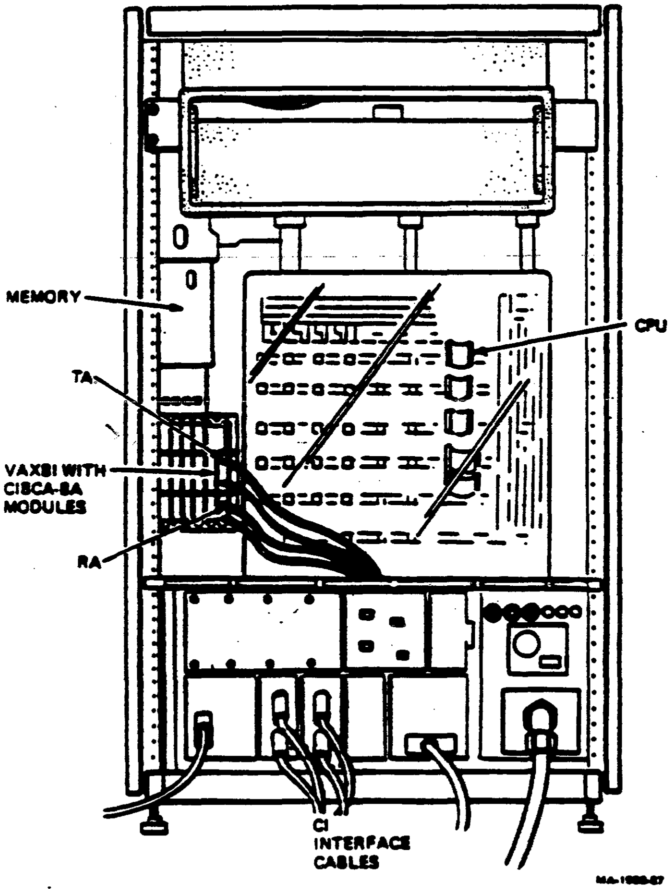
- The CIBCA modules occupy any two adjacent slots in the BI card cage.
- The "preferable" order is with the port controller module in the lower numbered slot. To the right
- The CIBCA-AA uses microcode file CIBCA.BIN which is loaded from the console media on system boot.
- The CIBCA-BA uses microcode file CIBCB.BIN which is resident on EEPROM so is not loaded at system boot. Use update utility EVGDA to manually load CIBCB.BIN from console into CIBCA interface.





Location of CIBCA Adapter (VAX 8500)

- o CI cables attach directly to the BI backplane.
- o BI node number determined by encapsulated plug.
- o CI node number determined by backplane jumpers (no switchpacks on CI adapter).
- o Backplane jumper assignment is the same for CIBCA-AA and CIBCA-BA.



CIBCA-BA Adapter in a VAX 8500 Series

CI PORT DIAGNOSTICS

CI Port Diagnostics

Lesson Introduction

Three major diagnostic tools are available for testing the CI Interface.

The repair diagnostics test each section of the hardware one at a time. After checking the condition of the Data Path, ALU, and microsequencer, small microcode routines are loaded into the control store and executed to test the rest of the individual pieces of the CI hardware. These require exclusive use of the CI Port and will call out the failing FRU. STAND Alone

Lowest Level

The functional diagnostics test the CI Port using the actual functional microcode. These are used to check the CI's ability to execute each of the functions that the standard microcode supports. They also require exclusive use of the CI Port.

The CI Exerciser is simply a verification of the CI Port's ability to communicate with a remote node. This exerciser will user the hardware and software that the operating system uses. It functions in conjunction with the operating system and does not interfere with any phase of the VAXcluster.

Lesson Objectives

1. State the purpose of each CI diagnostic and describe when each should be used.
2. Describe how to prepare for and run each CI diagnostic.
3. Verify the correct operation on the CI Port.
4. Verify the integrity of the CI Bus.
5. Identify the failing FRU in the CI Port.

Lesson Outline

I. Diagnostic Introduction
II. CI750 Repair Diagnostics
III. CI780 Repair Diagnostics
IV. CIBCI Repair Diagnostics
V. CI Functional Diagnostics
VI. CI Exerciser

NEW

CI DIAGNOSTIC SUMMARY

REPAIR DIAGNOSTICS (Level 3)

CI750 (750)	CI780 (780,86xx)	CIBCI (82xx,83xx,85xx,88xx)
ECCGA	EVCGA	EVCKA
ECCGB	EVCGB	EVCKB
ECCGC	EVCGC	EVCKC
ECCGD	EVCGD	EVCKD
ECCGE		EVCKE
		EVCKF
		EVCKG

CIBCA-AA (83xx,88xx)	CIBCA-BA (83xx,88xx,62xx)
EVGCA	EVGEE (EVGEK)
EVGCB (EVGCK)	EVGEF (EVGEL)
EVGCC (EVGCL)	EVGEG (EVGEM)
EVGCD (EVGCM)	EVGDA (CIBCB.BIN)
EVGCE (EVGCN)	

μCODE TAG ALONG FILES REQUIRED BY DIAGS

UPDATES EEPROM μCODE

- o CIBCA repair level diagnostics need repair level microcode to run properly (name of this microcode file is given in parenthesis).

- o Test the detailed hardware operation of the CI adapter.

FUNCTIONAL DIAGNOSTICS (Level 3) STAND alone

- o EVGAA and EVGAB are used on all CI interfaces.

- o Test the functional operation of the CI adapter when connected to a star coupler.

- ● Use of Event Flag 1 is important when running functionals.

☆ Clear = use microcode that is already loaded
set = load new microcode from load device

SET FLAG AFTER LOADING DIAG.

most have VMS

CI EXERCISER (Level 2R) EVXCI MUST BE INSTALLED ITS A SAVESET - NOT NORMALLY ON SYSTEM

- o Used to test communication betweeen nodes within a CI cluster.

- o Used to find a failing node or verify a repair.

Diagnostic Summary

Repair Diagnostics (Level 3)

CI750	CI780 (780,86xx)	CIBCI (82xx,83xx, 85xx,87xx,88xx)
ECCGA	EVCGA	EVCKA
ECCGB	EVCGB	EVCKB
ECCGC	EVCGC	EVCKC
ECCGD	EVCGD	EVCKD
ECCGE		EVCKE
		EVCKF
		EVCKG

VOID

Functional Diagnostics (Level 3)

- EVGAA and EVGAB are used on all CI Interfaces.
- Test the functional operation of the hardware and the microcode of CI Port.
- The use of Event Flag 1 is important when running functional diagnostics.

  Clear = use microcode that is already loaded
  Set = load new microcode from load device

CI Exerciser (Level 2R)

- Used to test communications between nodes.
- Used to find a failing node.
- Used to verify a repair.

Using the Autosizer on a CI Port

- Attaching the necessary devices in order to run repair or functional diagnostics on the CI Port can be facilitated by using the EVSBA/autosizer.

- **The user should verify that the node number found by the autosizer is the true node number. This is best done by running EVSBA in the Self-Test mode.**

VAX PAX VERSION 32 AND UP OK

```
DS> RUN EVSBA/SEC:SELFTEST

..  Program: EVSBA - AUTOSIZER LEVEL 3, revision 5.2, 3 tests, at 17:24:42.05.

COMMAND?  SIZE
!AUTOMATIC SIZING PROGRAM.
!THIS IS A TOOL AND NOT A DIAGNOSTIC PROGRAM.
         .

         .
DS>ATTACH KA86 HUB KA0 YES YES 1FFF 0
DS>ATTACH SBIA HUB SIO
DS>ATTACH DW780 SIO DW0 3 4 !  UNIBUS I/O SPACE NUMBER 0
         .                   BR
         .                  4 NODE
DS>ATTACH CI780 SIO PAA0 14    01
COMMAND?  CHANGE PAA0 NODE
ENTER VALUE: DECIMAL 0 THRU 255: 4
COMMAND?  ATTACH
COMMAND?  EXIT

...  End of run, 0 errors detected, pass count is 1, time is 17-jan-1987 17:26:25.14
```

VERIFY NODE # GIVEN BY AUTOSIZER IF WRONG - CHAGE

- Note that the order of commands entered by the user to change the node number was:

SIZE
CHANGE
ATTACH
EXIT
SELECT PAA0

Booting Into the Diagnostic Supervisor on a VAX-11/750

- To boot from a disk, insert the disk from which you are booting. From the console mode, type:

  ```
  >>> INITIALIZE
  >>> B/10 ddcu
  ```

  where:

  dd = the two character device code of the boot device

  c = the channel adapter to which the boot device is attached

  u = the drive number of the boot device

  For example:

  `>>> B/10 DMA0` -- boots from an RK07 on drive 0 of channel adapter A

  `>>> B/10 DBA0` -- boots from a MASSBUS disk on drive 0

- To boot from the console TU58, insert the appropriate tape cartridge. From the console mode, type:

  ```
  >>> INITIALIZE
  >>> B DDA0
  ```

  Once the Diagnostic Supervisor has been loaded and started it will identify itself and issue its user prompt:

  ```
  DS>
  ```

Repair-Level Diagnostics for the CI750

```
>>>B/10 DMA1
%%

DIAGNOSTIC SUPERVISOR.  ZZ-EXSAA- 6.10-323 15-MAR 1983 10:04:46
DS> ATTACH CI750 CMI PAA0 15 4 2
DS>        SELECT   PAA0
DS> SHOW SELECT
_PAA0   CI750   HUB   40F3E000    SLOT=15. BR=4.  Node-2.


DS> LOAD ECCGA
DS> SET TRACE
DS> START

.. Program CI750 - ECCGA Repair Level, revision 1.0, 13 tests,
   at 10:10:11.57.
Testing: _PAA0

Test 1: CNFGR REG - LONGWORD WRITE ACCESS
Test 2: CNFGR REG - LONGWORD READ ACCESS TEST
Test 3: CNFGR REG - LONGWORD READ/WRITE TEST
Test 4: CIPA TIME OUT TEST
Test 5: RCV FILE - ADDRESS TEST
Test 6: RCV FILE - DATA PATH TEST
Test 7: XMIT FILE - ADDRESS TEST
Test 8: XMIT FILE - DATA PATH TEST
Test 9: XMIT FILE - READ/WRITE COUNTER TEST
Test 10: CMMD ADDR REG - READ/WRITE AND DATA PATH TEST
Test 11: CMMD ADDR LO REG - READ INCREMENT TEST
Test 12: CIPA BUS - DATA PATH TEST
Test 13: RCV/XMIT FILE COUNTER SEQUENCE TEST
.. End of run, 0 errors detected, pass count is 1,
   time is 15-MAR-1983 10:10:25.39
DS>
```

RUNS
w/o
EXPANSION
BOX
PRESENT

NEEDS
EXP
BOX

ECCGA Printout

Repair-Level Diagnostics for the CI750 (Cont.)

- ECCGA
  - First of five repair-level diagnostics for the CI750.
  - Thirteen tests:

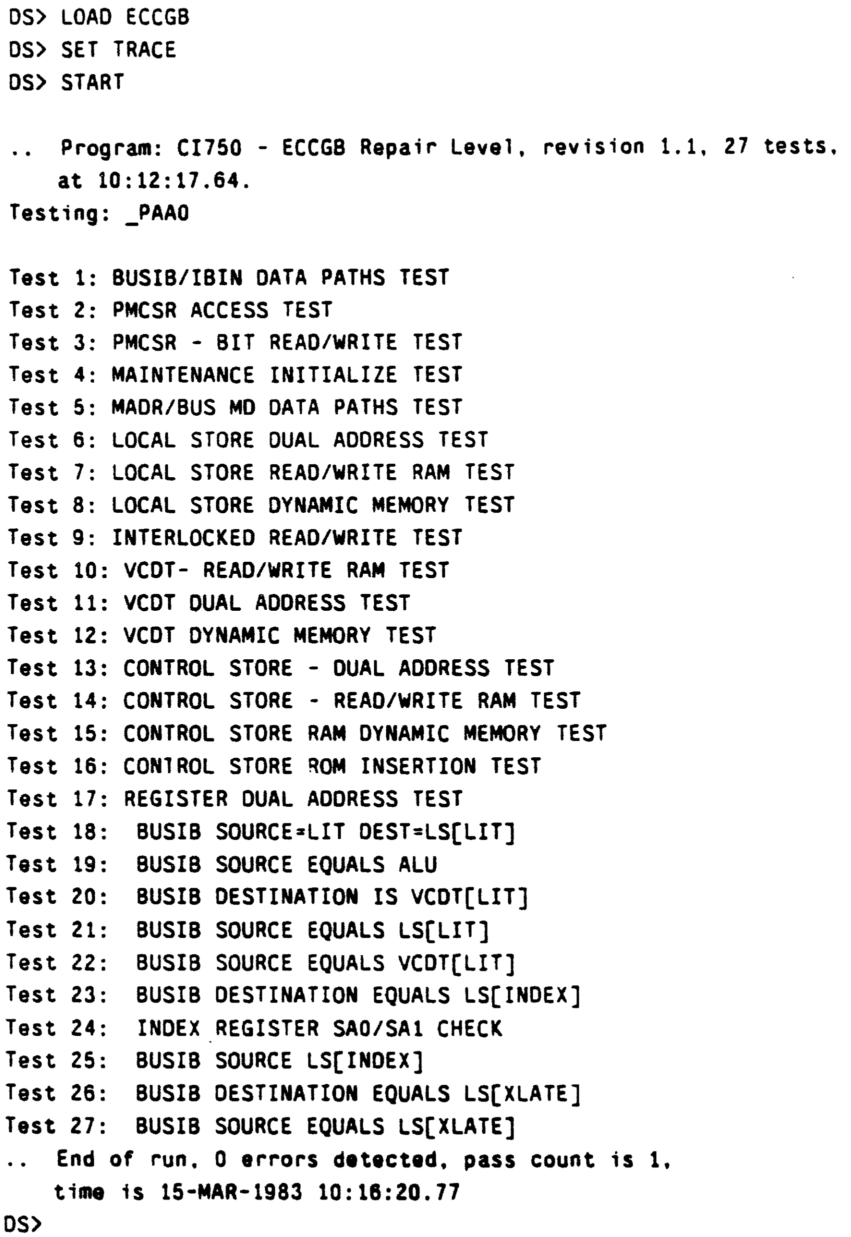
    Tests 1-11 will run without the CIPA present.
    Tests 12 and 13 require the CIPA.

- Sections

REGISTER	CNFGR, PMCSR, MADR, MDATR Registers

- Operation
  - Event Flag 5 is used to change the WCS tests for 3K.
  - QUICK is not implemented in this diagnostic.
  - Execution time for ECCGA is 3 seconds
  - No Summary report is issued by this program.

Repair-Level Diagnostics for the CI750 (Cont.)



```
DS> LOAD ECCGB
DS> SET TRACE
DS> START

.. Program: CI750 - ECCGB Repair Level, revision 1.1, 27 tests,
   at 10:12:17.64.
Testing: _PAA0

Test 1: BUSIB/IBIN DATA PATHS TEST
Test 2: PMCSR ACCESS TEST
Test 3: PMCSR - BIT READ/WRITE TEST
Test 4: MAINTENANCE INITIALIZE TEST
Test 5: MADR/BUS MD DATA PATHS TEST
Test 6: LOCAL STORE DUAL ADDRESS TEST
Test 7: LOCAL STORE READ/WRITE RAM TEST
Test 8: LOCAL STORE DYNAMIC MEMORY TEST
Test 9: INTERLOCKED READ/WRITE TEST
Test 10: VCDT- READ/WRITE RAM TEST
Test 11: VCDT DUAL ADDRESS TEST
Test 12: VCDT DYNAMIC MEMORY TEST
Test 13: CONTROL STORE - DUAL ADDRESS TEST
Test 14: CONTROL STORE - READ/WRITE RAM TEST
Test 15: CONTROL STORE RAM DYNAMIC MEMORY TEST
Test 16: CONTROL STORE ROM INSERTION TEST
Test 17: REGISTER DUAL ADDRESS TEST
Test 18:  BUSIB SOURCE=LIT DEST=LS[LIT]
Test 19:  BUSIB SOURCE EQUALS ALU
Test 20:  BUSIB DESTINATION IS VCDT[LIT]
Test 21:  BUSIB SOURCE EQUALS LS[LIT]
Test 22:  BUSIB SOURCE EQUALS VCDT[LIT]
Test 23:  BUSIB DESTINATION EQUALS LS[INDEX]
Test 24:  INDEX REGISTER SA0/SA1 CHECK
Test 25:  BUSIB SOURCE LS[INDEX]
Test 26:  BUSIB DESTINATION EQUALS LS[XLATE]
Test 27:  BUSIB SOURCE EQUALS LS[XLATE]
.. End of run, 0 errors detected, pass count is 1,
   time is 15-MAR-1983 10:16:20.77
DS>
```

ECCGB Printout

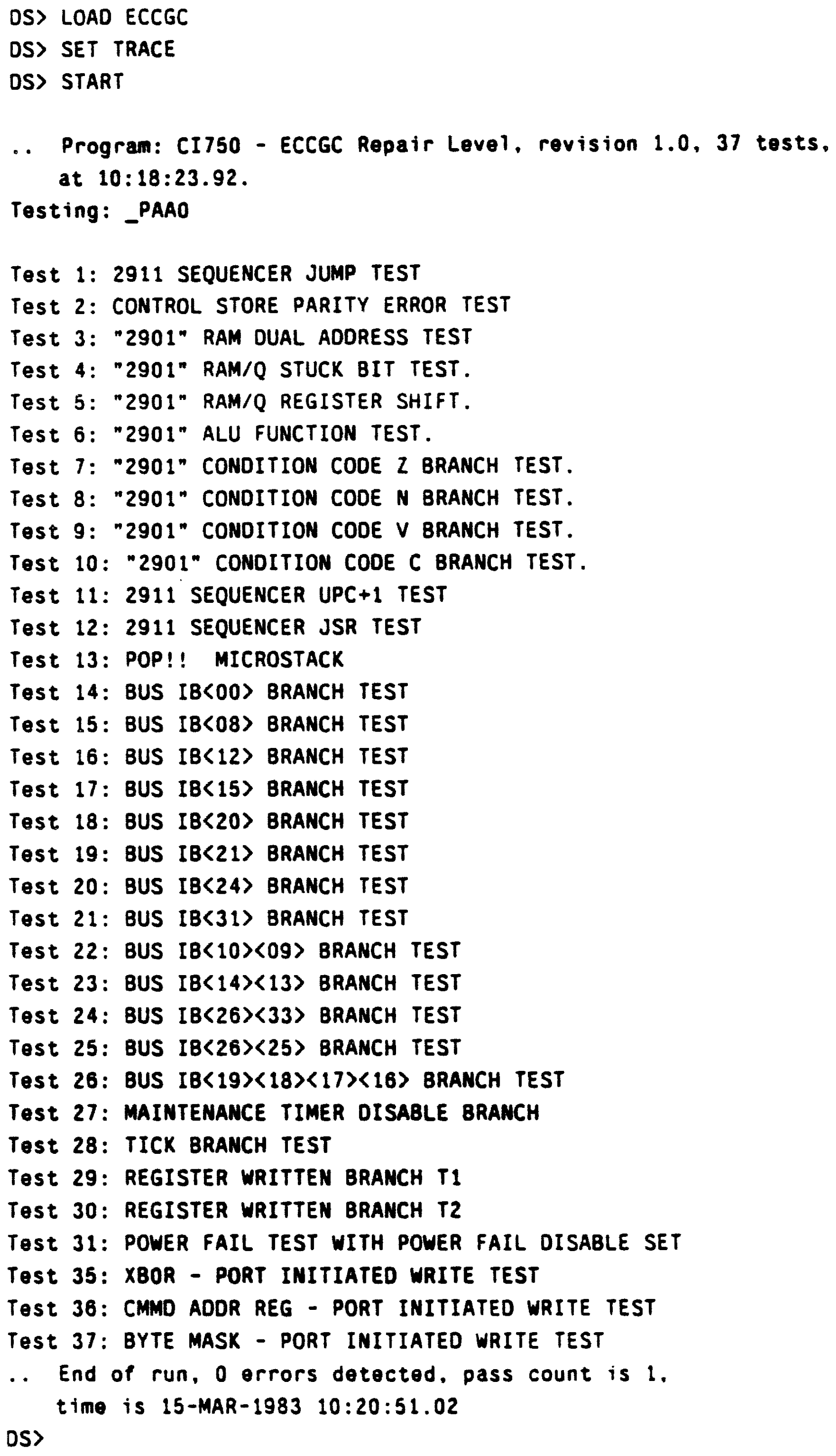
Repair-Level Diagnostics for the CI750 (Cont.)

- ECCGB
  - Second of five repair-level diagnostics for the CI750.
  - Twenty-seven tests.
- Sections

REGISTER	CNFGR, PMCSR, MADR, and MDATR Registers
LS__VCDT	Local Store/VCDT Tests
CONTROL__STORE	Control Store Tests
IB__SRC__DST	Bus IB Source and Destination Tests

- Operation
  - Event Flag 5 is used to change the WCS tests for 3K.
  - QUICK is not implemented in this diagnostic.
  - Execution time for ECCGB is 4 minutes.
  - No Summary report is issued by this program.

Repair-Level Diagnostics for the CI750 (Cont.)



```
DS> LOAD ECCGC
DS> SET TRACE
DS> START

..  Program: CI750 - ECCGC Repair Level, revision 1.0, 37 tests,
    at 10:18:23.92.
Testing: _PAA0

Test 1: 2911 SEQUENCER JUMP TEST
Test 2: CONTROL STORE PARITY ERROR TEST
Test 3: "2901" RAM DUAL ADDRESS TEST
Test 4: "2901" RAM/Q STUCK BIT TEST.
Test 5: "2901" RAM/Q REGISTER SHIFT.
Test 6: "2901" ALU FUNCTION TEST.
Test 7: "2901" CONDITION CODE Z BRANCH TEST.
Test 8: "2901" CONDITION CODE N BRANCH TEST.
Test 9: "2901" CONDITION CODE V BRANCH TEST.
Test 10: "2901" CONDITION CODE C BRANCH TEST.
Test 11: 2911 SEQUENCER UPC+1 TEST
Test 12: 2911 SEQUENCER JSR TEST
Test 13: POP!!  MICROSTACK
Test 14: BUS IB<00> BRANCH TEST
Test 15: BUS IB<08> BRANCH TEST
Test 16: BUS IB<12> BRANCH TEST
Test 17: BUS IB<15> BRANCH TEST
Test 18: BUS IB<20> BRANCH TEST
Test 19: BUS IB<21> BRANCH TEST
Test 20: BUS IB<24> BRANCH TEST
Test 21: BUS IB<31> BRANCH TEST
Test 22: BUS IB<10><09> BRANCH TEST
Test 23: BUS IB<14><13> BRANCH TEST
Test 24: BUS IB<26><33> BRANCH TEST
Test 25: BUS IB<26><25> BRANCH TEST
Test 26: BUS IB<19><18><17><16> BRANCH TEST
Test 27: MAINTENANCE TIMER DISABLE BRANCH
Test 28: TICK BRANCH TEST
Test 29: REGISTER WRITTEN BRANCH T1
Test 30: REGISTER WRITTEN BRANCH T2
Test 31: POWER FAIL TEST WITH POWER FAIL DISABLE SET
Test 35: XBOR - PORT INITIATED WRITE TEST
Test 36: CMMD ADDR REG - PORT INITIATED WRITE TEST
Test 37: BYTE MASK - PORT INITIATED WRITE TEST
..  End of run, 0 errors detected, pass count is 1,
    time is 15-MAR-1983 10:20:51.02
DS>
```

ECCGC Printout

Repair-Level Diagnostics for the CI750 (Cont.)

- ECCGC
  - Third of five repair-level diagnostics for the CI750.
  - Thirty-seven tests:

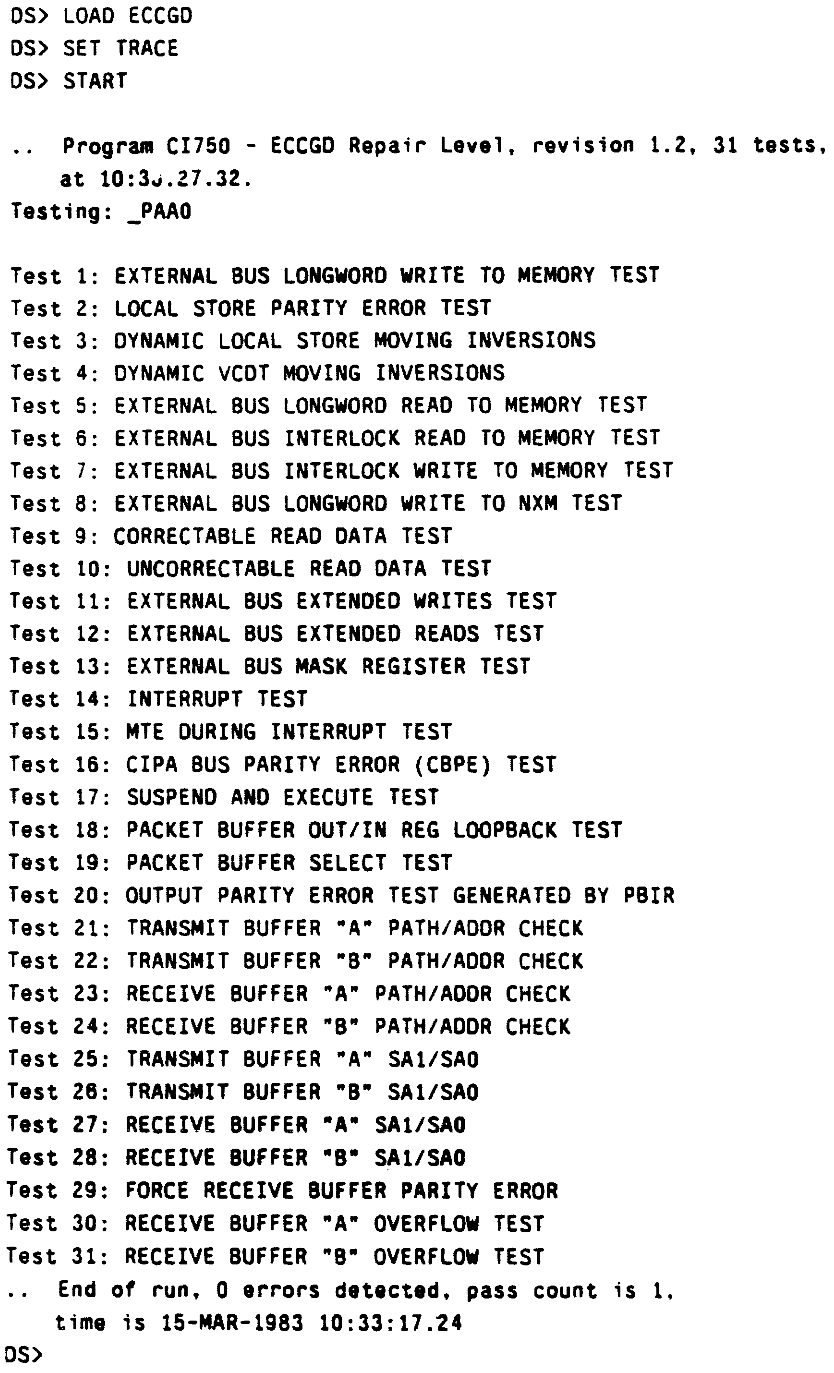
    Tests 1-31 and 35-37 run in the default section.

    Tests 32-34 are Manual Intervention Tests.
- Sections

SEQUENCER	2911 Sequencer Tests
ALU	2901 ALU Tests
BRANCH	Microcode Branch Tests
MANUAL	Manual Intervention Test

- Event Flags
  - Flag 5 is used to change the WCS tests for 3K.
  - Flag 6 is used for microlooping. <- FOR MANUFACTURING USE !
- Operation
  - QUICK is not implemented in this diagnostic.
  - Execution time for ECCGC is 2 minutes and 28 seconds (excluding the Manual Intervention Test).
  - No Summary report is issued by this program.

Repair-Level Diagnostics for the CI750 (Cont.)



```
DS> LOAD ECCGD
DS> SET TRACE
DS> START

..  Program CI750 - ECCGD Repair Level, revision 1.2, 31 tests,
    at 10:3ȷ.27.32.
Testing: _PAA0

Test 1: EXTERNAL BUS LONGWORD WRITE TO MEMORY TEST
Test 2: LOCAL STORE PARITY ERROR TEST
Test 3: DYNAMIC LOCAL STORE MOVING INVERSIONS
Test 4: DYNAMIC VCDT MOVING INVERSIONS
Test 5: EXTERNAL BUS LONGWORD READ TO MEMORY TEST
Test 6: EXTERNAL BUS INTERLOCK READ TO MEMORY TEST
Test 7: EXTERNAL BUS INTERLOCK WRITE TO MEMORY TEST
Test 8: EXTERNAL BUS LONGWORD WRITE TO NXM TEST
Test 9: CORRECTABLE READ DATA TEST
Test 10: UNCORRECTABLE READ DATA TEST
Test 11: EXTERNAL BUS EXTENDED WRITES TEST
Test 12: EXTERNAL BUS EXTENDED READS TEST
Test 13: EXTERNAL BUS MASK REGISTER TEST
Test 14: INTERRUPT TEST
Test 15: MTE DURING INTERRUPT TEST
Test 16: CIPA BUS PARITY ERROR (CBPE) TEST
Test 17: SUSPEND AND EXECUTE TEST
Test 18: PACKET BUFFER OUT/IN REG LOOPBACK TEST
Test 19: PACKET BUFFER SELECT TEST
Test 20: OUTPUT PARITY ERROR TEST GENERATED BY PBIR
Test 21: TRANSMIT BUFFER "A" PATH/ADDR CHECK
Test 22: TRANSMIT BUFFER "B" PATH/ADDR CHECK
Test 23: RECEIVE BUFFER "A" PATH/ADDR CHECK
Test 24: RECEIVE BUFFER "B" PATH/ADDR CHECK
Test 25: TRANSMIT BUFFER "A" SA1/SA0
Test 26: TRANSMIT BUFFER "B" SA1/SA0
Test 27: RECEIVE BUFFER "A" SA1/SA0
Test 28: RECEIVE BUFFER "B" SA1/SA0
Test 29: FORCE RECEIVE BUFFER PARITY ERROR
Test 30: RECEIVE BUFFER "A" OVERFLOW TEST
Test 31: RECEIVE BUFFER "B" OVERFLOW TEST
..  End of run, 0 errors detected, pass count is 1,
    time is 15-MAR-1983 10:33:17.24
DS>
```

ECCGD Printout

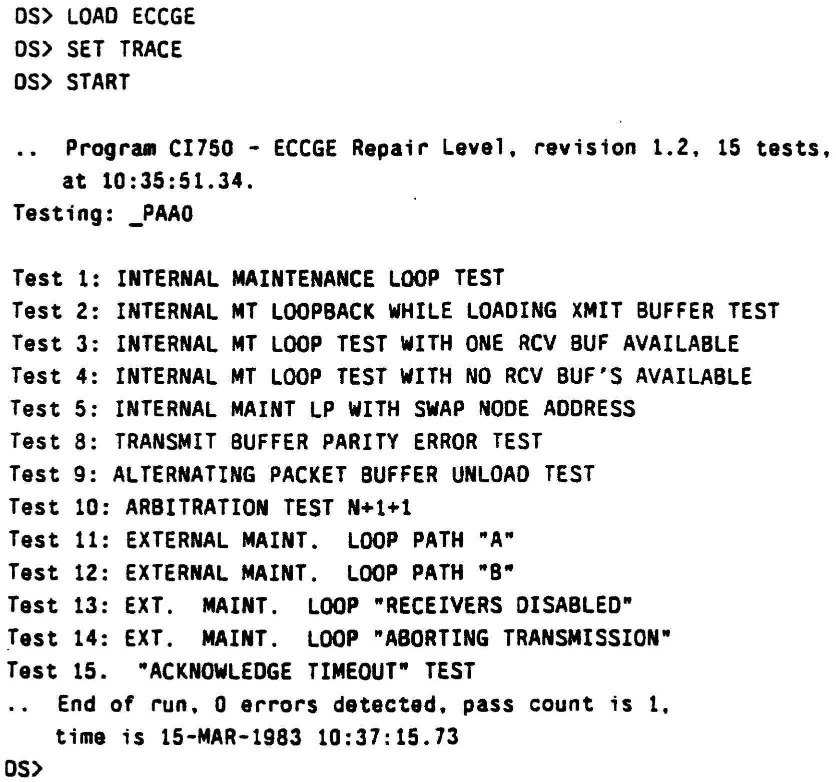
Repair-Level Diagnostics for the CI750 (Cont.)

- ECCGD
  - Fourth of five repair-level diagnostics for the CI750.
  - Thirty-one tests.
- Sections

EXTBUF	External Bus Tests (CMI)
PBUFFER	Packet Buffer Tests

- Event Flags -- Flag 6 is used for microlooping. ← USED BY MANUFACTURING
- Operation
  - QUICK is not implemented in this diagnostic.
  - Execution time for ECCGD is 2 minutes and 24 seconds.
  - No Summary report is issued by this program.

Repair-Level Diagnostics for the CI750 (Cont.)



```
 DS> LOAD ECCGE
 DS> SET TRACE
 DS> START

 ..  Program CI750 - ECCGE Repair Level, revision 1.2, 15 tests,
     at 10:35:51.34.
 Testing: _PAA0

 Test 1: INTERNAL MAINTENANCE LOOP TEST
 Test 2: INTERNAL MT LOOPBACK WHILE LOADING XMIT BUFFER TEST
 Test 3: INTERNAL MT LOOP TEST WITH ONE RCV BUF AVAILABLE
 Test 4: INTERNAL MT LOOP TEST WITH NO RCV BUF'S AVAILABLE
 Test 5: INTERNAL MAINT LP WITH SWAP NODE ADDRESS
 Test 8: TRANSMIT BUFFER PARITY ERROR TEST
 Test 9: ALTERNATING PACKET BUFFER UNLOAD TEST
 Test 10: ARBITRATION TEST N+1+1
 Test 11: EXTERNAL MAINT.  LOOP PATH "A"
 Test 12: EXTERNAL MAINT.  LOOP PATH "B"
 Test 13: EXT.  MAINT.  LOOP "RECEIVERS DISABLED"
 Test 14: EXT.  MAINT.  LOOP "ABORTING TRANSMISSION"
 Test 15.  "ACKNOWLEDGE TIMEOUT" TEST
 ..  End of run, 0 errors detected, pass count is 1,
     time is 15-MAR-1983 10:37:15.73
 DS>
```

ECCGE Printout

Repair-Level Diagnostics for the CI750 (Cont.)

- ECCGE
  - Fifth of five repair-level diagnostics for the CI750.
  - Fifteen tests: RE START [illegible]

    Tests 1-5 and 8-15 run in the default section.
    Tests 6-7 run only in the manual intervention section. PROMPT TO CHANGE ADDR SWITCHES

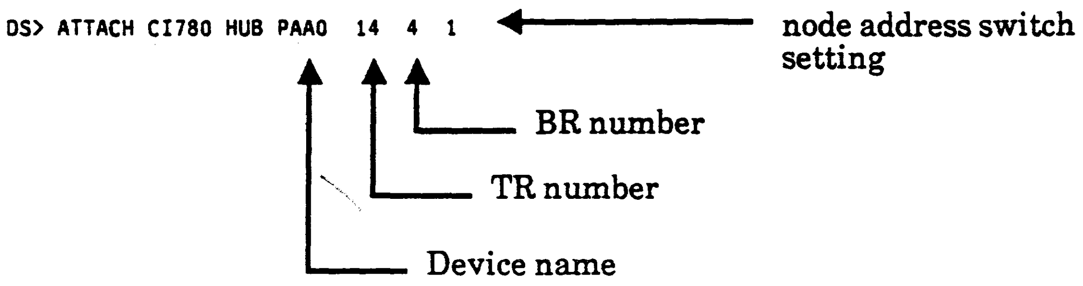
- Sections

INT__MLOOP	Internal Maintenance Loop Tests
EXT__MLOOP	External Maintenance Loop Tests
MANUAL	Manual Intervention Tests

- Event Flags
  - Flag 6 is used for microlooping.
  - Flag 7 should be set if the extended-header jumper is installed. This affects Test 15 (Acknowledge Timeout Test).
  - Flag 8 should be set if the extended-acknowledge-timeout jumper is installed. This affects Test 15 (Acknowledge Timeout Test).
  - Flag 10, when set, allows as many as 64 no-response retries when running Tests 11 and 12 (External Maintenance Loop Path Tests). SET IF RUNNING ON BUSY CLUSTER SO TRAFFIC DOESN'T MAKE DIAG TIMEOUT

- Operation
  - QUICK feature is not implemented in this diagnostic.
  - Execution time for ECCGE is 40 seconds (excluding Manual Intervention Tests).
  - No Summary report is issued by this program.

Attaching Devices on a VAX-11/780



```
DS> ATTACH CI780 HUB PAA0  14  4  1
```

Attaching Devices on a VAX 8600

```
DS> ATTACH SBIA HUB SI0
DS> ATTACH CI780 SI0 PAA0 14 4 1
```

Booting in the Diagnostic Supervisor on a VAX-11/780

- When booting from a disk, insert the standard console floppy diskette and the disk from which you are booting, and type:

  ```
  >>> BOOT Sgn
  ```

  where:

  S = Diagnostic boot

  g = Generic drive type (B for an RP drive, R for an RM drive, M for an RK drive)

  n = Drive number

  For example:

  ```
  >> BOOT SBO - boots from drive unit 0 of an RP06
  ```

- When booting from the console floppy, insert the load path floppy diskette, and from console mode type:

  ```
  >>> HALT
  >>> INITIALIZE
  >>> UNJAM
  >>> LOAD/START:FE00 ESSAA.EXE
  >>> START 10000
  ```

  GOOD!
  USE THIS DIAG IF FLOPPY ISN'T BOOTABLE!

  Once the VDS has been loaded and started it will identify itself and issue its user prompt:

  ```
  DS>
  ```

Booting in the Diagnostic Supervisor on a VAX 8600

- When booting from a non-console disk, insert the standard console RL02 disk and the disk from which you are booting and type:

  ```
  >>> BOOT/R5:10 <device>
  ```

  where:

  device = a one to three character mnemonic (such as DU0 or CS1) that is appended to BOO.COM to form a file name that will be used to boot the operating system; if no device is specified, DEFBOO.COM will be used

  For example:

  `>>> BOOT/R5:10 DU0` -- loads R5 with 10 and invokes DU0BOO.COM

  `>>> BOOT/R5:10/NOSTART` -- loads R5, runs DEFBOO.COM, waits for further input

- When booting from a console disk, enter the following command:

  ```
  >>> @EDSAA
  ```

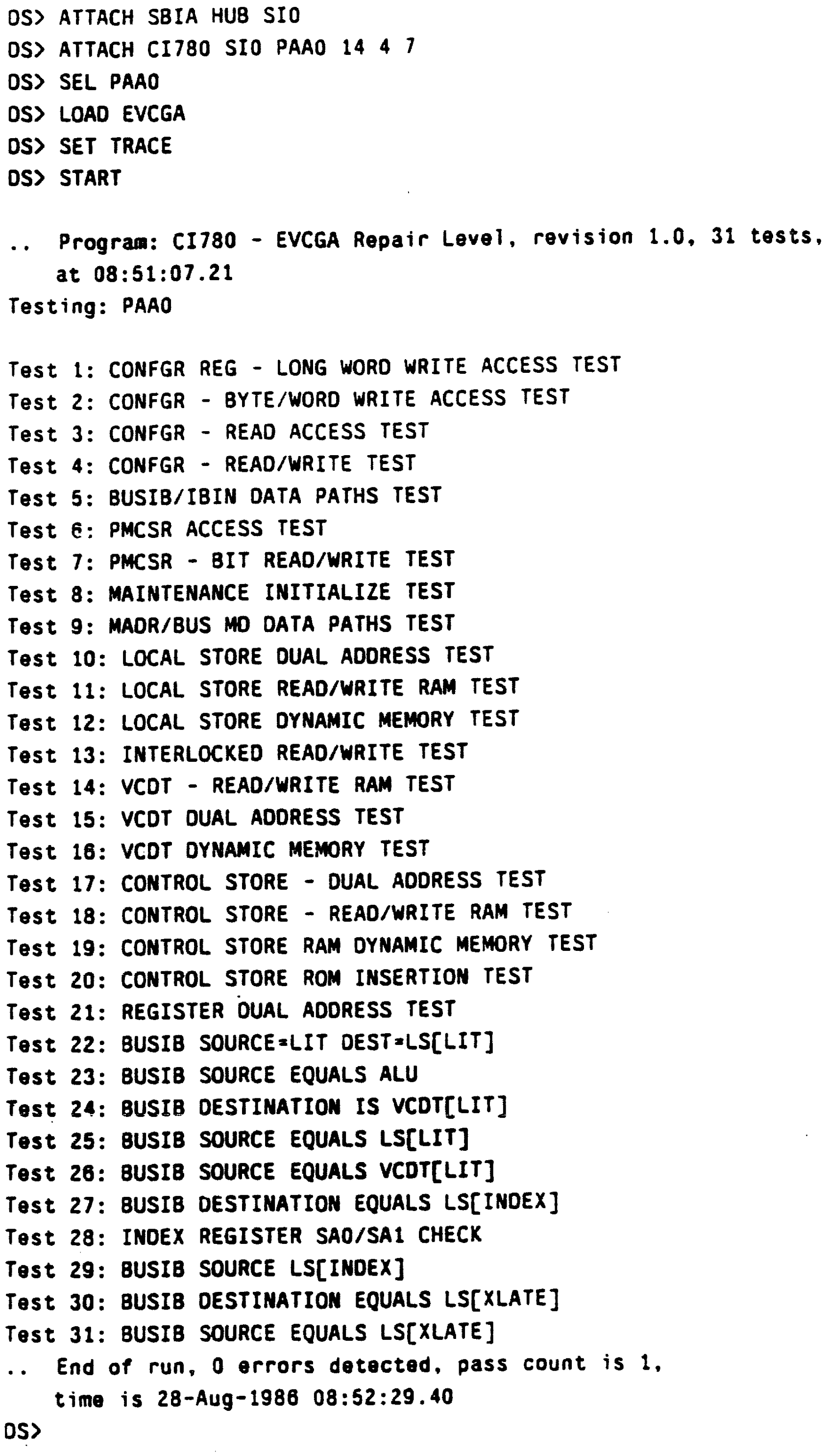
  Once the VDS has been loaded and started, it will identify itself and issue its user prompt:

  ```
  DS>
  ```

>>SET QUIET OFF } (PRINTS BOOT FILE DURING BOOT)
>>B

>>> D R5 XXXXXX10

Repair-Level Diagnostics for the CI780



```
DS> ATTACH SBIA HUB SIO
DS> ATTACH CI780 SIO PAAO 14 4 7
DS> SEL PAAO
DS> LOAD EVCGA
DS> SET TRACE
DS> START

..  Program: CI780 - EVCGA Repair Level, revision 1.0, 31 tests,
    at 08:51:07.21
Testing: PAAO

Test 1: CONFGR REG - LONG WORD WRITE ACCESS TEST
Test 2: CONFGR - BYTE/WORD WRITE ACCESS TEST
Test 3: CONFGR - READ ACCESS TEST
Test 4: CONFGR - READ/WRITE TEST
Test 5: BUSIB/IBIN DATA PATHS TEST
Test 6: PMCSR ACCESS TEST
Test 7: PMCSR - BIT READ/WRITE TEST
Test 8: MAINTENANCE INITIALIZE TEST
Test 9: MADR/BUS MD DATA PATHS TEST
Test 10: LOCAL STORE DUAL ADDRESS TEST
Test 11: LOCAL STORE READ/WRITE RAM TEST
Test 12: LOCAL STORE DYNAMIC MEMORY TEST
Test 13: INTERLOCKED READ/WRITE TEST
Test 14: VCDT - READ/WRITE RAM TEST
Test 15: VCDT DUAL ADDRESS TEST
Test 16: VCDT DYNAMIC MEMORY TEST
Test 17: CONTROL STORE - DUAL ADDRESS TEST
Test 18: CONTROL STORE - READ/WRITE RAM TEST
Test 19: CONTROL STORE RAM DYNAMIC MEMORY TEST
Test 20: CONTROL STORE ROM INSERTION TEST
Test 21: REGISTER DUAL ADDRESS TEST
Test 22: BUSIB SOURCE=LIT DEST=LS[LIT]
Test 23: BUSIB SOURCE EQUALS ALU
Test 24: BUSIB DESTINATION IS VCDT[LIT]
Test 25: BUSIB SOURCE EQUALS LS[LIT]
Test 26: BUSIB SOURCE EQUALS VCDT[LIT]
Test 27: BUSIB DESTINATION EQUALS LS[INDEX]
Test 28: INDEX REGISTER SA0/SA1 CHECK
Test 29: BUSIB SOURCE LS[INDEX]
Test 30: BUSIB DESTINATION EQUALS LS[XLATE]
Test 31: BUSIB SOURCE EQUALS LS[XLATE]
..  End of run, 0 errors detected, pass count is 1,
    time is 28-Aug-1986 08:52:29.40
DS>
```

EVCGA Printout

Repair-Level Diagnostics for the CI780 (Cont.)

- EVCGA
  - First of four repair-level diagnostics for the CI780.
  - Thirty-one tests.
- Sections

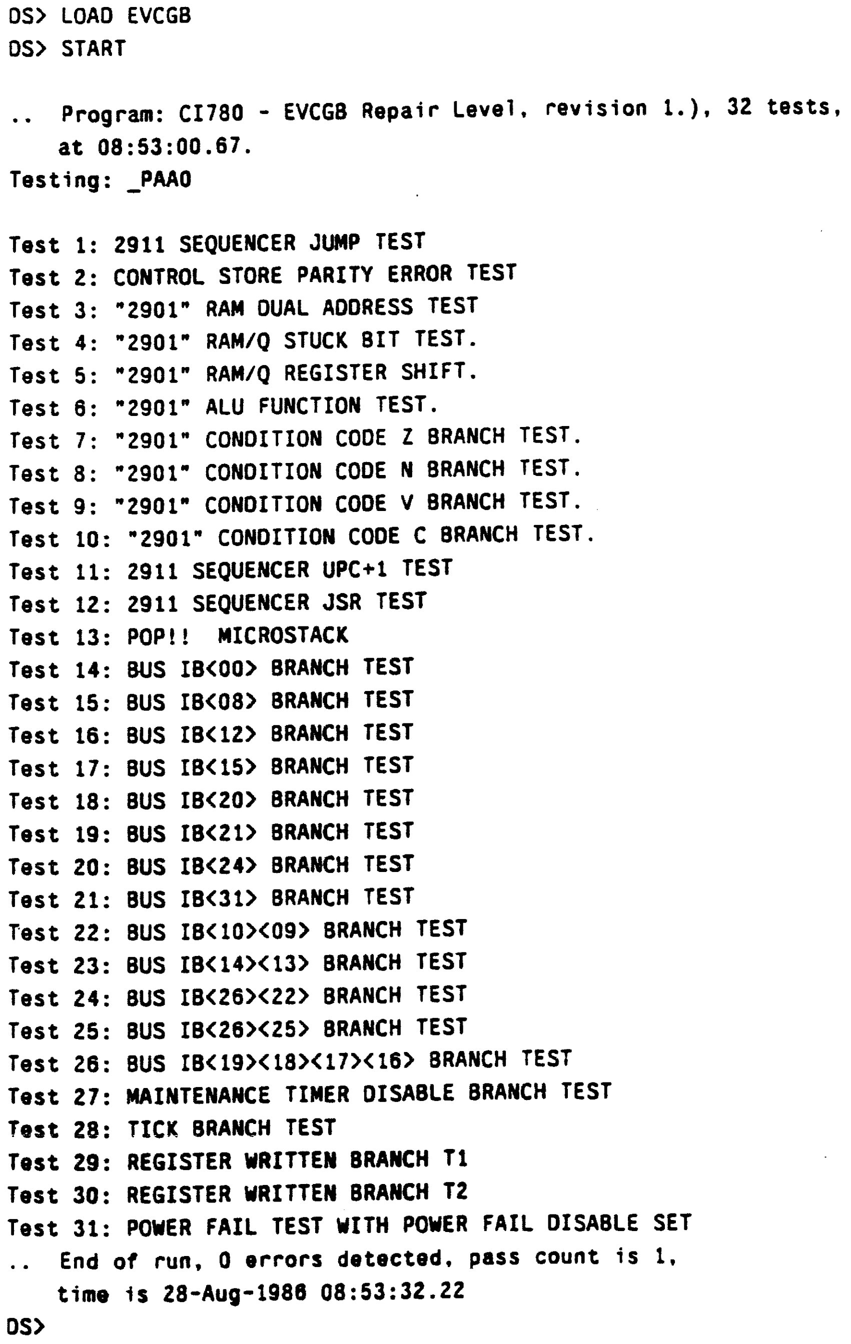
REGISTER	CNFGR, PMCSR, MADR, MDATR Registers
LS__VCDT	Local Store/VCDT Tests
CONTROL__STORE	Control Store Tests
IB__SRC__DST	Bus IB Source and Destination Tests

- Operation

  writable control store

  - Event Flag 5 is used to change the WCS tests for 3K. ONLY 2K PRESENT will ERR IF SET
  - QUICK is not implemented in this diagnostic.
  - Execution time for EVCGA is 1 minute, 30 seconds.
  - No Summary report is issued by this program.

Repair-Level Diagnostics for the CI780 (Cont.)



```
DS> LOAD EVCGB
DS> START

.. Program: CI780 - EVCGB Repair Level, revision 1.), 32 tests,
   at 08:53:00.67.
Testing: _PAA0

Test 1: 2911 SEQUENCER JUMP TEST
Test 2: CONTROL STORE PARITY ERROR TEST
Test 3: "2901" RAM DUAL ADDRESS TEST
Test 4: "2901" RAM/Q STUCK BIT TEST.
Test 5: "2901" RAM/Q REGISTER SHIFT.
Test 6: "2901" ALU FUNCTION TEST.
Test 7: "2901" CONDITION CODE Z BRANCH TEST.
Test 8: "2901" CONDITION CODE N BRANCH TEST.
Test 9: "2901" CONDITION CODE V BRANCH TEST.
Test 10: "2901" CONDITION CODE C BRANCH TEST.
Test 11: 2911 SEQUENCER UPC+1 TEST
Test 12: 2911 SEQUENCER JSR TEST
Test 13: POP!!  MICROSTACK
Test 14: BUS IB<00> BRANCH TEST
Test 15: BUS IB<08> BRANCH TEST
Test 16: BUS IB<12> BRANCH TEST
Test 17: BUS IB<15> BRANCH TEST
Test 18: BUS IB<20> BRANCH TEST
Test 19: BUS IB<21> BRANCH TEST
Test 20: BUS IB<24> BRANCH TEST
Test 21: BUS IB<31> BRANCH TEST
Test 22: BUS IB<10><09> BRANCH TEST
Test 23: BUS IB<14><13> BRANCH TEST
Test 24: BUS IB<26><22> BRANCH TEST
Test 25: BUS IB<26><25> BRANCH TEST
Test 26: BUS IB<19><18><17><16> BRANCH TEST
Test 27: MAINTENANCE TIMER DISABLE BRANCH TEST
Test 28: TICK BRANCH TEST
Test 29: REGISTER WRITTEN BRANCH T1
Test 30: REGISTER WRITTEN BRANCH T2
Test 31: POWER FAIL TEST WITH POWER FAIL DISABLE SET
.. End of run, 0 errors detected, pass count is 1,
   time is 28-Aug-1986 08:53:32.22
DS>
```

EVCGB Printout

Repair-Level Diagnostics for the CI780 (Cont.)

- EVCGB
  - Second of four repair-level diagnostics for the CI780.
  - Thirty-two tests:

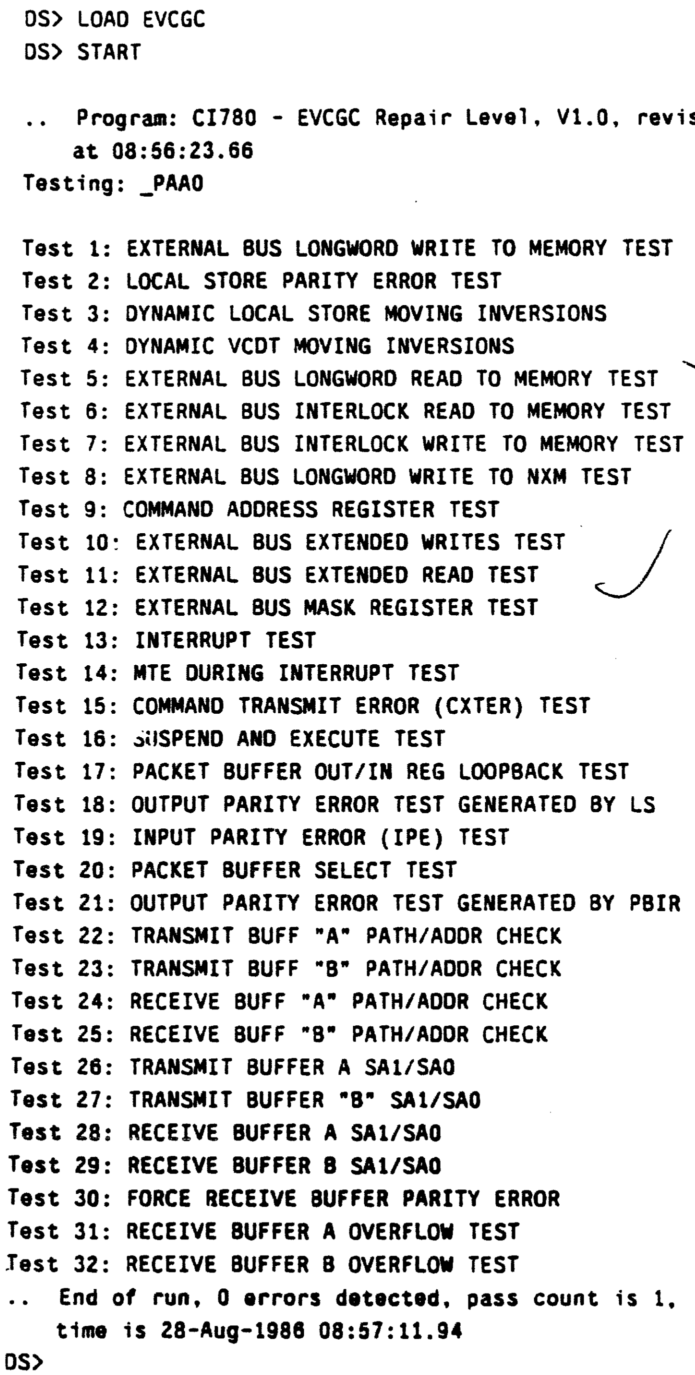
    Tests 1-31 run in the default section.
    Test 32 runs only in the manual section.

- Sections

SEQUENCER	2911 Sequencer Tests
ALU	2901 ALU Tests
BRANCH	Microcode Branch Tests
MANUAL	Manual Intervention Test

- Event Flags
  - Flag 5 is used to change the WCS tests for 3K.
  - Flag 6 is used for microlooping.
- Operation
  - QUICK is not implemented in this diagnostic.
  - Execution time for EVCGB is 52 seconds (excluding the Manual Intervention Test).
  - No Summary report is issued by this program.

Repair-Level Diagnostics for the CI780 (Cont.)



```
DS> LOAD EVCGC
DS> START

..  Program: CI780 - EVCGC Repair Level, V1.0, revision 1.0, 32 tests,
    at 08:56:23.66
Testing: _PAA0

Test 1: EXTERNAL BUS LONGWORD WRITE TO MEMORY TEST
Test 2: LOCAL STORE PARITY ERROR TEST
Test 3: DYNAMIC LOCAL STORE MOVING INVERSIONS
Test 4: DYNAMIC VCDT MOVING INVERSIONS
Test 5: EXTERNAL BUS LONGWORD READ TO MEMORY TEST
Test 6: EXTERNAL BUS INTERLOCK READ TO MEMORY TEST
Test 7: EXTERNAL BUS INTERLOCK WRITE TO MEMORY TEST
Test 8: EXTERNAL BUS LONGWORD WRITE TO NXM TEST
Test 9: COMMAND ADDRESS REGISTER TEST
Test 10: EXTERNAL BUS EXTENDED WRITES TEST
Test 11: EXTERNAL BUS EXTENDED READ TEST
Test 12: EXTERNAL BUS MASK REGISTER TEST
Test 13: INTERRUPT TEST
Test 14: MTE DURING INTERRUPT TEST
Test 15: COMMAND TRANSMIT ERROR (CXTER) TEST
Test 16: SUSPEND AND EXECUTE TEST
Test 17: PACKET BUFFER OUT/IN REG LOOPBACK TEST
Test 18: OUTPUT PARITY ERROR TEST GENERATED BY LS
Test 19: INPUT PARITY ERROR (IPE) TEST
Test 20: PACKET BUFFER SELECT TEST
Test 21: OUTPUT PARITY ERROR TEST GENERATED BY PBIR
Test 22: TRANSMIT BUFF "A" PATH/ADDR CHECK
Test 23: TRANSMIT BUFF "B" PATH/ADDR CHECK
Test 24: RECEIVE BUFF "A" PATH/ADDR CHECK
Test 25: RECEIVE BUFF "B" PATH/ADDR CHECK
Test 26: TRANSMIT BUFFER A SA1/SA0
Test 27: TRANSMIT BUFFER "B" SA1/SA0
Test 28: RECEIVE BUFFER A SA1/SA0
Test 29: RECEIVE BUFFER B SA1/SA0
Test 30: FORCE RECEIVE BUFFER PARITY ERROR
Test 31: RECEIVE BUFFER A OVERFLOW TEST
Test 32: RECEIVE BUFFER B OVERFLOW TEST
..  End of run, 0 errors detected, pass count is 1,
    time is 28-Aug-1986 08:57:11.94
DS>
```

EXT BUS
= SBI
NOT CI !

EVCGC Printout

Repair-Level Diagnostics for the CI780 (Cont.)

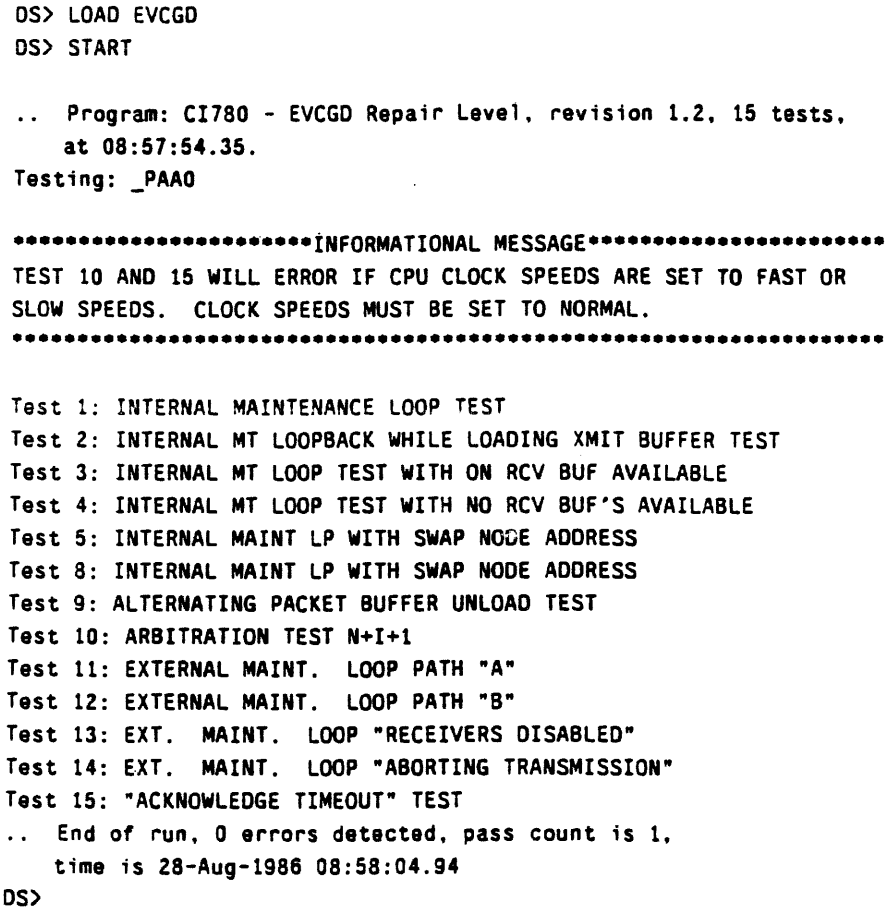
- EVCGC
  - Third of four repair-level diagnostics for the CI780.
  - ~~Thirty tests~~. 32 TESTS
- Sections

  (SBI) BUS

EXTBUS	External Bus Tests (SBI)
PBUFFER	Packet Buffer Tests

- Event Flag 6 is used for microlooping. MANUF. ONLY
- Operation
  - QUICK is not implemented in this diagnostic.
  - Execution time for EVCGC is 1 minute, 19 seconds.
  - No Summary report is issued by this program.

Repair-Level Diagnostics for the CI780 (Cont.)



```
DS> LOAD EVCGD
DS> START

..  Program: CI780 - EVCGD Repair Level, revision 1.2, 15 tests,
    at 08:57:54.35.
Testing: _PAA0

**********************INFORMATIONAL MESSAGE*********************
TEST 10 AND 15 WILL ERROR IF CPU CLOCK SPEEDS ARE SET TO FAST OR
SLOW SPEEDS.  CLOCK SPEEDS MUST BE SET TO NORMAL.
****************************************************************

Test 1: INTERNAL MAINTENANCE LOOP TEST
Test 2: INTERNAL MT LOOPBACK WHILE LOADING XMIT BUFFER TEST
Test 3: INTERNAL MT LOOP TEST WITH ON RCV BUF AVAILABLE
Test 4: INTERNAL MT LOOP TEST WITH NO RCV BUF'S AVAILABLE
Test 5: INTERNAL MAINT LP WITH SWAP NODE ADDRESS
Test 8: INTERNAL MAINT LP WITH SWAP NODE ADDRESS
Test 9: ALTERNATING PACKET BUFFER UNLOAD TEST
Test 10: ARBITRATION TEST N+I+1
Test 11: EXTERNAL MAINT.  LOOP PATH "A"
Test 12: EXTERNAL MAINT.  LOOP PATH "B"
Test 13: EXT.  MAINT.  LOOP "RECEIVERS DISABLED"
Test 14: EXT.  MAINT.  LOOP "ABORTING TRANSMISSION"
Test 15: "ACKNOWLEDGE TIMEOUT" TEST
..  End of run, 0 errors detected, pass count is 1,
    time is 28-Aug-1986 08:58:04.94
DS>
```

EVCGD Printout

Repair-Level Diagnostics for the CI780 (Cont.)

- EVCGD
  - Fourth of four repair-level diagnostics for the CI780.
  - Fifteen tests:

    Tests 1-5, 8, 9, and 11-14 run in the default section.
    Tests 6, 7, [redacted] run only in the manual section.

TEST LINK BRD. SWITCHES

set flag 10

- Sections

INT__MLOOP	Internal Maintenance Loop Tests
EXT__MLOOP	External Maintenance Loop Tests
MANUAL	Manual Intervention Tests

- Event Flags
  - Flag 6 is used for microlooping.
  - Flag 7 should be set if the extended header jumper is installed. This affects Test 15 (Acknowleage Timeout Test).
  - Flag 8 should be set if the extended acknowledge timeout jumper is installed. This affects Test 15 (Acknowledge Timeout Test).
  - Flag 10, when set, will allow as many as 64 no-response retries when running Tests 11 and 12 (External Maintenance Loop Path Tests). SET IF ON BUSY CI
- Operation
  - QUICK feature is not implemented in this diagnostic.
  - Execution time for EVCGD is 13 seconds (excluding Manual Intervention Tests).
  - No SUMMARY report is issued by this program.

Loading the Diagnostic Supervisor on a VAX 8200/8300

- Insert the RX50 diskette containing file EBSAA.EXE into the console RX50 disk drive unit 0.

- Load the diagnostic supervisor program into physical memory by entering the following CCL command at the console terminal:

```
>>> B/R5:10 CSA1:
```

  or

```
>>> B/R5:10 DU40 ←—— Drive unit number
               ↑
               └—— BI node number
```

- Identify the CIBCI adapter and its node configuration parameters to the diagnostic supervisor program as follows:

```
DS> ATTACH CIBCI HUB PAA0 6 4 0
```

- Select the CIBCI adapter:

```
DS> SELECT PAA0
```

- Show the unit selected as follows:

```
DS> SHOW SELECT
```

Loading the Diagnostic Supervisor on a VAX 85xx/8700/8800

- Insert the RX50 diskette containing file EZSAA.EXE into the PC380 console system RX50 disk drive unit 0.

- Load the diagnostic supervisor program resident on the PC380 console system disk into physical memory by entering the following CCL command. Use the PC380 console system keyboard as follows:

```
>>> @DIABOO
%%
DIAGNOSTIC SUPERVISORP .1e1a;DS>
```

- Attach the CPU to its memory and the NBI adapters as follows:

```
                               = PRIMARY CPU
DS> ATTACH KAAAA HUB KA0 YES          CPU 1
DS> ATTACH KAAAA HUB KA1 NO           CPU 2
                            NOT PRIM
DS> ATTACH MSAAA HUB MS0              MEM
DS> ATTACH NBIA HUB NBIA0 0           ADAPT
DS> ATTACH NBIB NBIA0 NBIB0 0 0       BUS
DS> ATTACH NBIB NBIA0 NBIB1 1 0       ADAPT
DS> ATTACH NBIA HUB NBIA1 1           BUS
DS> ATTACH NBIB NBIA1 NBIB0 0 0
DS> ATTACH NBIB NBIA1 NBIB1 1 0
```

- Identify the CIBCI adapter and its node configuration parameters to the diagnostic supervisor:

```
DS> ATTACH CIBCI NBIB0 PAA0 6 4 0
```

- Select the CIBCI adapter as the unit under test:

```
DS> SELECT PAA0
```

- Show the unit selected:

```
DS> SHOW SELECT
```

Repair-Level Diagnostics for the CIBCI

```
DS> LOAD EVCKA
DS> SET FLAGS TRACE, HALT
DS> SET EVENT FLAGS 4
DS> START/PASS:5

..  Program: CIBCI - EVCKA Repair level, revision 1.0, 28 tests,
    at 00:24:40.75.
    Testing: _PAA0

SET EVENT FLAG 4 FOR REV LEVEL OF BIIC IN TEST 3

Test 1: ERROR INTERRUPT CONTROL TEST
Test 2: DEVICE TYPE REGISTER TEST
Test 3: BC AND CIBCI SELF TEST
REVISION LEVEL OF BIIC CHIP ON CIBCI IS: 0

Test 4: CNFGR - L WRITE ACCESS TEST
Test 5: CNFGR - L READ ACCESS TEST
Test 6: R/W TEST OF DIAG BIT IN CNFGR
Test 7: CNFGR - L READ ACCESS TEST - AFTER DISABLING UCSREEN IN BCICR
Test 8: CNFGR - L READ ACCESS TEST - AFTER DISABLING STS IN BCICR
Test 9: PORT DATA REGISTER - R/W TEST - SOURCE IS B1
Test 10: R/W TEST OF BUFFERED COMMAND ADDRESS REGISTER (BCAR)
Test 11: R/W TEST OF BCMR
Test 12: R/W TEST OF DMA REGISTER
Test 13: RECEIVED COMMAND DATA PATH TEST
Test 14: R/W TEST OF CNFGR, BCAR AND BCMR TAKEN ALTOGETHER
Test 15: SIZE OF TRANSFER TEST
Test 16: DMA FILE - R/W COUNTER TEST
Test 17: DMA FILE - COUNTER SEQUENCE TEST
Test 18: R/W TEST OF BCAR AND BCMR USING THE MASTER SEQUENCER
Test 19: BICA ADDRESS REGISTER TEST
Test 20: STOP TEST
Test 21: PORT DATA REGISTER - CIPA DATA PATH TEST
Test 22: WITH DIAG BIT CLEAR, R/W TEST OF BCAR
Test 23: WITH DIAG BIT CLEAR, R/W TEST OF DMA REGISTER
Test 24: CIPAPD REGISTER READ TEST (CIPA BUX READ TEST)
Test 25: L READ ACCESS TEST OF LS AFTER DISABLING UCSREEN IN BCI CONTROL REG
Test 26: NUACK TEST FOR NODE ADDRESS 200
Test 27: L READ ACCESS TEST OF LS AFTER DISABLING STS IN THE BICSR
Test 28: USER INTERRUPT CONTROL TEST

..  End of run, 0 errors detected, pass count is 1,
    time is 15-JUL-1985 00:24:52.74
DS>
```

Trace Printout for Repair Diagnostic EVCKA

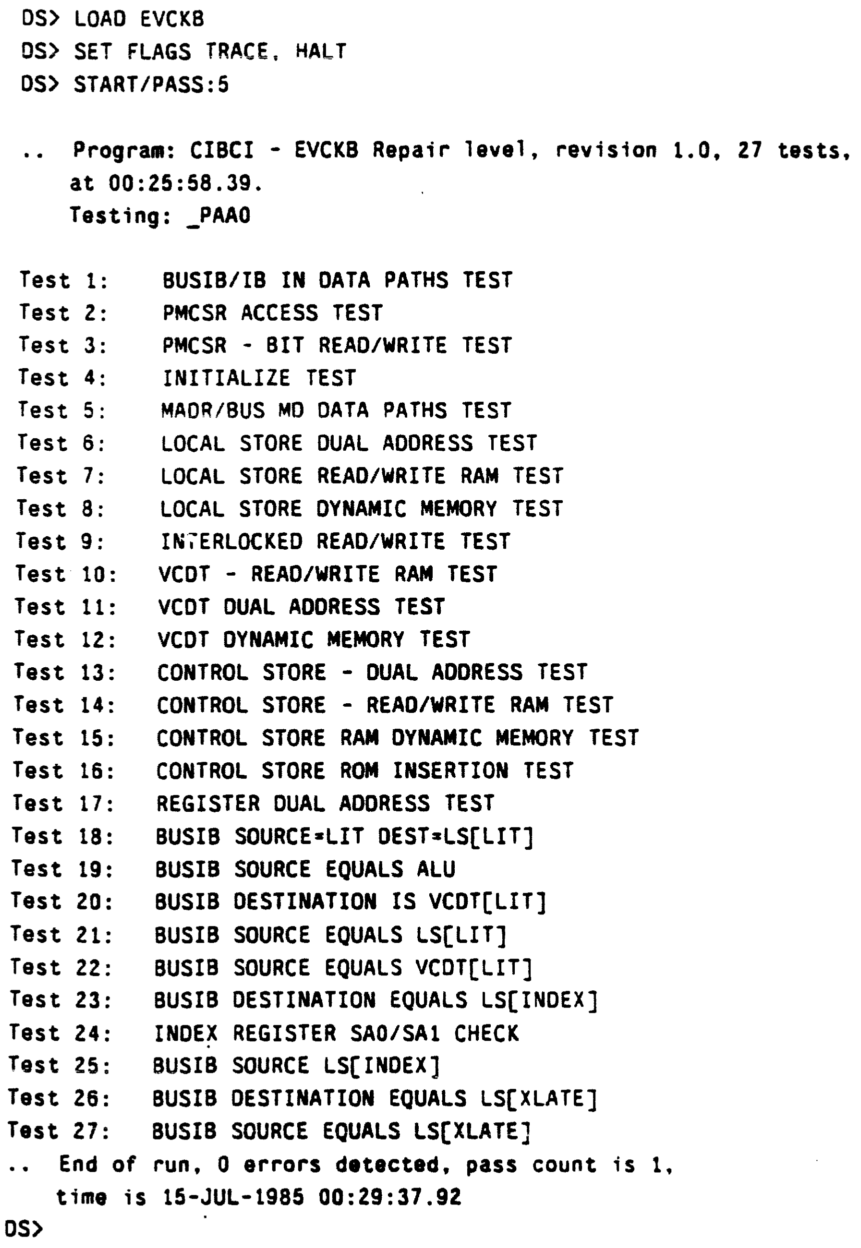
Repair-Level Diagnostics for the CIBCI (Cont.)

- EVCKA
  - First of six repair-level diagnostics for the CIBCI.
  - Twenty-eight tests.
- Sections

BAC	BICA Control module
BAD	BICA Data module
CIPA	CIPA box

- Event Flags
  - Flag 4 is used to output revision level of BIIC chip in Test 3.
  - Flag 5 determines whether the message pertaining to the use of Event Flag 4 will get printed at the start of the diagnostic.
- Operation
  - QUICK feature not implemented in this diagnostic.
  - Execution time is approximately 12 seconds.
  - The summary report provides an error count by test number, no report is generated if there are no errors.

Repair-Level Diagnostics for the CIBCI (Cont.)



```
DS> LOAD EVCKB
DS> SET FLAGS TRACE, HALT
DS> START/PASS:5

..  Program: CIBCI - EVCKB Repair level, revision 1.0, 27 tests,
    at 00:25:58.39.
    Testing: _PAA0

Test 1:    BUSIB/IB IN DATA PATHS TEST
Test 2:    PMCSR ACCESS TEST
Test 3:    PMCSR - BIT READ/WRITE TEST
Test 4:    INITIALIZE TEST
Test 5:    MADR/BUS MD DATA PATHS TEST
Test 6:    LOCAL STORE DUAL ADDRESS TEST
Test 7:    LOCAL STORE READ/WRITE RAM TEST
Test 8:    LOCAL STORE DYNAMIC MEMORY TEST
Test 9:    INTERLOCKED READ/WRITE TEST
Test 10:   VCDT - READ/WRITE RAM TEST
Test 11:   VCDT DUAL ADDRESS TEST
Test 12:   VCDT DYNAMIC MEMORY TEST
Test 13:   CONTROL STORE - DUAL ADDRESS TEST
Test 14:   CONTROL STORE - READ/WRITE RAM TEST
Test 15:   CONTROL STORE RAM DYNAMIC MEMORY TEST
Test 16:   CONTROL STORE ROM INSERTION TEST
Test 17:   REGISTER DUAL ADDRESS TEST
Test 18:   BUSIB SOURCE=LIT DEST=LS[LIT]
Test 19:   BUSIB SOURCE EQUALS ALU
Test 20:   BUSIB DESTINATION IS VCDT[LIT]
Test 21:   BUSIB SOURCE EQUALS LS[LIT]
Test 22:   BUSIB SOURCE EQUALS VCDT[LIT]
Test 23:   BUSIB DESTINATION EQUALS LS[INDEX]
Test 24:   INDEX REGISTER SA0/SA1 CHECK
Test 25:   BUSIB SOURCE LS[INDEX]
Test 26:   BUSIB DESTINATION EQUALS LS[XLATE]
Test 27:   BUSIB SOURCE EQUALS LS[XLATE]
..  End of run, 0 errors detected, pass count is 1,
    time is 15-JUL-1985 00:29:37.92
DS>
```

Trace Printout for Repair Diagnostic EVCKB

Repair-Level Diagnostics for the CIBCI (Cont.)

- EVCKB
  - Second of six repair-level diagnostics for the CIBCI.
  - Twenty-seven tests.
- Sections

REGISTER	CNFGR, PMCSR, MADR, MDATR Registers
LS__VCDT	Local Store/VCDT Tests
CONTROL__STORE	Control Store Tests
IB__SRC__DST	Bus IB Source and Dest Tests

- Event Flag 5 is used to change the WCS test for 3K.
- Operation
  - QUICK feature not implemented.
  - No Summary report is issued.

Repair-Level Diagnostics for the CIBCI (Cont.)

```
DDS> LOAD EVCKC
DS> SET FLAGS TRACE. HALT
DS> START/PASS:5

..  Program: CIBCI - EVCKC Repair level, revision 1.0, 33 tests,
    at 00:30:00.96
    Testing: _PAA0

    Test 1:     2911 SEQUENCER JUMP TEST
    Test 2:     CONTROL STORE PARITY ERROR TEST
    Test 3:     "2901" RAM DUAL ADDRESS TEST
    Test 4:     "2901" RAM/Q STUCK BIT TEST
    Test 5:     "2301" RAM/Q REGISTER SHIFT
    Test 6:     "2901" ALU FUNCTION TEST.
    Test 7:     "2901" CONDITION CODE Z BRANCH TEST.
    Test 8:     "2901" CONDITION CODE N BRANCH TEST.
    Test 9:     "2901" CONDITION CODE V BRANrH TEST.
    Test 10:    "2901" CONDITION CODE C BRANCH TEST.
    Test 11:    2911 SEQUENCER UPC+1 TEST
    Test 12:    2911 SEQUENCER JSR TEST
    Test 13:    POP!!  MICROSTCK
    Test 14:    BUS IB<00> BRANCH TEST
    Test 15:    BUS IB<08> BRANCH TEST
    Test 16:    BUS IB<12> BRANCH TEST
    Test 17:    BUS IB<15> BRANCH TEST
    Test 18:    BUS IB<20> BRANCH TEST
    Test 19:    BUS IB<21> BRANCH TEST
    Test 20:    BUS IB<24> BRANCH TEST
    Test 21:    BUS IB<31> BRANCH TEST
    Test 22:    BUS IB<10> <09> BRANCH TEST
    Test 23:    BUS IB<14> <13> BRANCH TEST
    Test 24:    BUS IB<26> <22> BRANCH TEST
    Test 25:    BUS IB<26> <25> BRANCH TEST
    Test 26:    BUS IB<19> <18> <17> <16> BRANCH TEST
    Test 27:    MAINTENANCE TIMER DISABLE BRANCH TEST
    Test 28:    TICK BRANCH TEST
    Test 29:    REGISTER WRITTEN BRANCH T1
    Test 30:    REGISTER WRITTEN BRANCH T2
    Test 31:    XBOR - PORT INITIATED WRITE TEST
    Test 32:    BICA CMMD ADDR REG - PORT INITIATED WRITE TEST
    Test 33:    BYTE MASK - PORT INITIATED WRITE TEST
..  End of run, 0 errors detected, pass count is 1,
    time is 15-JUL-1985 00:32:56:24
DS>
```

Trace Printout for Repair Diagnostic EVCKC

Repair-Level Diagnostics for the CIBCI (Cont.)

- EVCKC:
  - Third of six repair-level diagnostics for the CIBCI.
  - Thirty-three tests.
- Sections

SEQUENCER	2911 Sequencer Tests
ALU	2901 ALU Tests
BRANCH	Microcode Branch Tests

- Event Flags
  - Event Flag 5 used to change the WCS test for 3K.
  - Event Flag 6 used for microlooping.
- Operation
  - QUICK feature not implemented.
  - No Summary report is issued.
  - Execution time is 3 minutes, 52 seconds.

Repair-Level Diagnostics for the CIBCI (Cont.)

```
DS> LOAD EVCKD
DS> SET FLAGS TRACE, HALT
DS> START/PASS:5

..Program: CIBCI - EVCKD Repair level, revision 1.0, 21 tests,
  at 00:33:15.20.
  Testing: _PAA0

  Test  1:    EXTERNAL BUS LONGWORD WRITE TO MEMORY TEST
  Test  2:    LOCAL STORE PARITY ERROR TEST
  Test  3:    DYNAMIC LOCAL STORE MOVING INVERSIONS
  Test  4:    DYNAMIC VCDT MOVING INVERSIONS
  Test  5:    EXTERNAL BUS LONGWORD READ TO MEMORY TEST
  Test  6:    EXTERNAL BUS INTERLOCK READ TO MEMORY TEST
  Test  7:    EXTERNAL BUS INTERLOCK WRITE TO MEMORY TEST
  Test  8:    EXTERNAL BUS LONGWORD WRITE TO NXM TEST
  Test  9:    CORRECTABLE READ DATA TEST FOR VAX-11/750
  TEST  IGNORED FOR THIS PROCESSOR

  Test  10:   READ DATA SUBSTITUTE TEST FOR VAX-11/750
  TEST  IGNORED FOR THIS PROCESSOR

  Test  11:   READ DATA SUBSTITUTE TEST FOR VAX 8700
  TEST  IGNORED FOR THIS PROCESSOR

  Test  12:   CORRECTABLE READ DATA TEST FOR VAX 8200
  Test  11:   READ DATA SUBSTITUTE TEST FOR VAX 8200
  Test  14:   EXTERNAL BUS EXTENDED WRITES TEST
  Test  15:   EXTERNAL BUS EXTENDED READS TEST
  Test  16:   EXTERNAL BUS MASK REGISTER TEST
  Test  17:   INTERRUPT TEST
  Test  18:   MTE DURING INTERRUPT TEST
  Test  19:   CIPA BUS PARITY ERROR (CBPE) TEST
  Test  20:   SUSPEND AND EXECUTE TEST
  Test  21:   PACKET BUFFER UUT/IN REG LOOPBACK TEST
..End of run, 0 errors detected, pass count is 1,
  time is 15-JUL-1985 00:34:43.62
DS>
```

Trace Printout for Repair Diagnostic EVCKD

Repair-Level Diagnostics for the CIBCI (Cont.)

- EVCKD
  - Fourth of seven repair-level diagnostics for the CIBCI.
  - Twenty-one tests:

    Tests 1-8 and 14-21 run in the default mode.

    Tests 9 and 10 pertain to tests used in early development work and can be ignored.

    Test 11 pertains to the RDS test for VAX 8800.

    Test 12 and 13 pertain to CRD and RDS tests for the VAX 8200/8300.

- Sections

EXTBUS	External Bus Tests (BI)
PBUFFER	Packet Buffer Tests
CBA	CRD and RDS Test pertaining to the CBA Test bed
NAUTILUS	RDS Test pertaining to VAX 8800
SCORPIO	CRD and RDS Tests pertaining to VAX 8200/8300

- Event Flags
  - Event Flag 1 controls the output of the message indicating that a test will not be performed because of CPU incompatibility (tests 9, 10, 11, 12).
  - Event Flag 6 is used for microlooping. MAN

- Operation
  - QUICK feature not implemented.
  - No Summary report is issued.
  - Execution time is 2 minutes, 7 seconds.

Repair-Level Diagnostics for the CIBCI (Cont.)

```
DS> LOAD EVCKE
DS> SET FLAGS TRACE, HALT
DS> START/PASS:5

..Program: CIBCI - EVCKE Repair level, revision 1.0, 13 tests,
  at 00:34:59.99.
  Testing: _PAA0

  Test  1:    PACKET BUFFER SELECT TEST
  Test  2:    OUTPUT PARITY ERROR TEST GENERATED BY PBIR
  Test  3:    TRANSMIT BUFFER "A" PATH/ADDR CHECK
  Test  4:    TRANSMIT BUFFER "B" PATH/ADDR CHECK
  Test  5:    RECEIVE BUFFER "A" PATH/ADDR CHECK
  Test  6:    RECEIVE BUFFER "B" PATH/ADDR CHECK
  Test  7:    TRANSMIT BUFFER "A" SA1/SA0
  Test  8:    TRANSMIT BUFFER "B" SA1/SA0
  Test  9:    RECEIVE BUFFER "A" SA1/SA0
  Test  10:   RECEIVE BUFFER "B" SA1/SDA0
  Test  11:   FORCE RECEIVE BUFFER PARITY ERROR
  Test  12:   RECEIVE BUFFER "A" OVERFLOW TEST
  Test  13:   RECEIVE BUFFER "B" OVERFLOW TEST
..End of run, 0 errors detected, pass count is 1,
  time is 15-JUL-1985 00:36:29.06
DS>
```

Trace Printout for Repair Diagnostic EVCKE

Repair-Level Diagnostics for the CIBCI (Cont.)

- EVCKE
  - Fifth of seven repair-level diagnostics for the CIBCI.
  - Nineteen tests:

    Tests 14, 15, 16, 17, 18 and 19 are run in manual mode.
    Test 14 is a true node address test.
    Test 15 is a complement node address test.
    Test 16 is a boot time test.
    Test 17 is a CIPA power fail test.
    Test 18 and 19 are CPU power fail test.

- Sections

PBUFFER	Packet Buffer Test
MANUAL	Manual Intervention Tests

- Event Flag 6 is used for microlooping.
- Operation
  - QUICK feature not implemented.
  - No Summary report is issued.
  - Execution time is 2 minutes, 40 seconds.

Repair-Level Diagnostics for the CIBCI (Cont.)

```
DS> LOAD EVCKF
DS> SET FLAGS TRACE, HALT
DS> START/PASS:5

..Program: CIBCI - EVCKF Repair level, revision 1.0, 14 tests,
  at 00:39:31.19.
  Testing: _PAA0

  Test  1:    INTERNAL MAINTENANCE LOOP TEST
  Test  2:    INTERNAL MT LOOPBACK WHILE LOADING XMIT BUFFER TEST
  Test  3:    INTERNAL MT LOOP TEST WITH ONE RCV BUF AVAILABLE
  Test  4:    INTERNAL MT LOOP TEST WITH NO REV BUF'S AVAILABLE
  Test  5:    INTERNAL MAINT LP WITH SWAP NODE ADDRESS
  Test  6:    TRANSMIT BUFFER PARITY ERROR TEST
  Test  7:    ALTERNATING PACKET BUFFER UNLOAD TEST
  Test  8:    ARBITRATION TEST N+I+1
```

Trace Printout for Repair Diagnostic EVCKF

Repair-Level Diagnostics for the CIBCI (Cont.)

- EVCKF
  - Sixth of seven repair-level diagnostics for the CIBCI.
  - Eight tests.
- Section

INT__MLOOP	Internal Maintenance Loop Tests

- Event Flags
  - Flag 6 is used for microlooping.
  - Flag 11 is used for quiet slot value test of link module.
- Operation
  - QUICK feature not implemented.
  - No Summary report is issued.
  - Execution time is 35 seconds.

Repair-Level Diagnostics for the CIBCI (Cont.)

```
DS> LOAD EVCKG
DS> SET FLAGS TRACE, HALT
DS> START/PASS:5

..Program: CIBCI - EVCDG Repair level, revision 1.0, 6 tests
  at 00:39:31.19
  Testing: _PAA0

  Test  1:      EXTERNAL MAINT.  LOOP PATH "A"
  Test  2:      EXTERNAL MAINT.  LOOP PATH "B"
  Test  3:      EXT.  MAINT.  LOOP "RECEIVERS DISABLED"
  Test  4:      EXT.  MAINT.  LOOP "ABORTING TRANSMISSION"
  Test  5:      "ACKNOWLEDGE TIMEOUT" TEST
  Test  6:      EXTERNAL BUS LONGWORD WRITE TO ITSELF (LOCAL STORE)
..End of run, 0 errors detected, pass count is 1,
  time is 15-JUL-1985 00:40:10.22
```

Trace Printout for Repair Diagnostic EVCKG

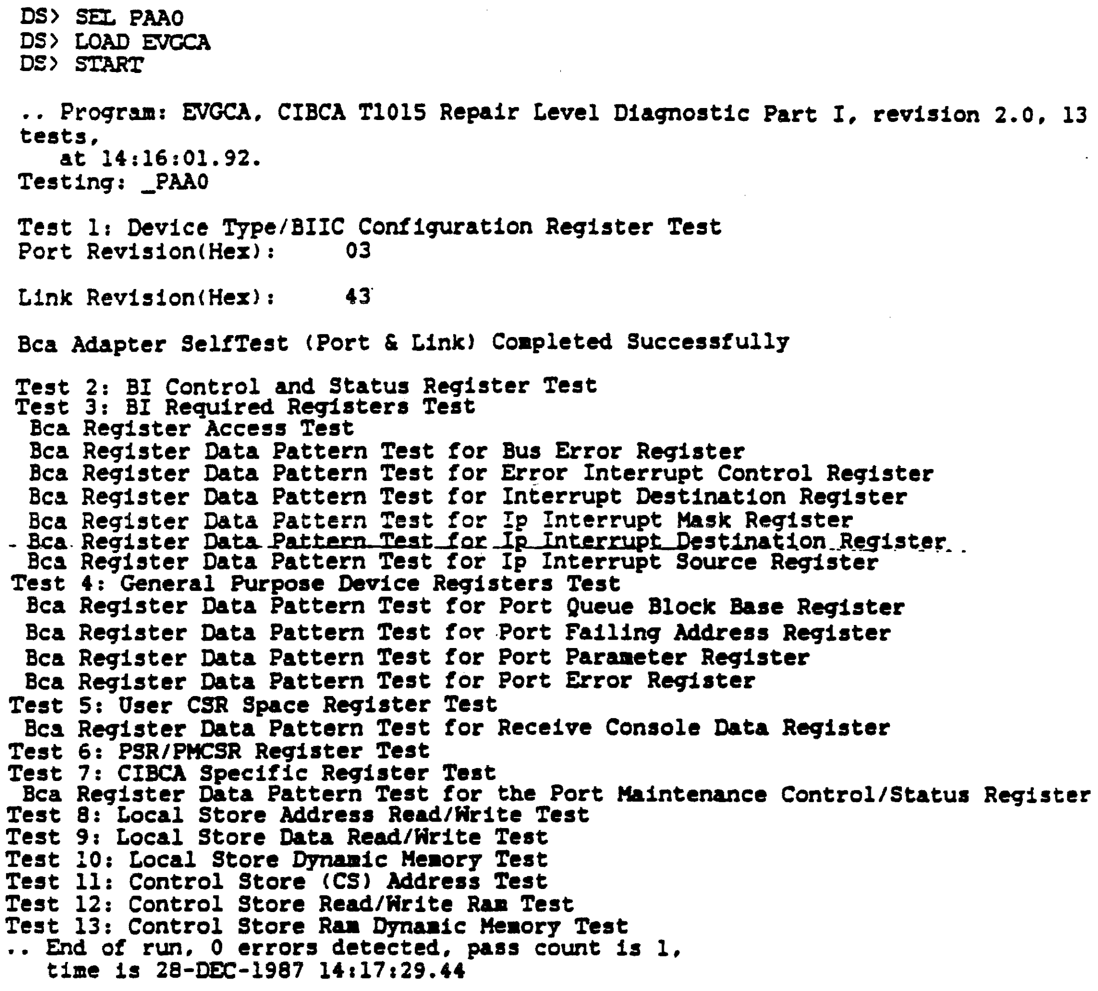
Repair-Level Diagnostics for the CIBCI (Cont.)

- EVCKG
  - Seventh of seven repair-level diagnostics for the CIBCI.
  - Six tests.
- Section

EXT_MLOOP	External Maintentance Loop Tests

- Event Flags
  - Flag 6 is used for microlooping.
  - Flag 7 is used for extended header.
  - Flag 8 is used for extended ACK timeout (test 5).
  - Flag 10 allows up to 64 no-response retries when running tests 1 and 2.
- Operation
  - QUICK feature not implemented.
  - No Summary report is issued.
  - Execution time is 35 seconds.

Repair-Level Diagnostics for the CIBCA



```
DS> SEL PAA0
DS> LOAD EVGCA
DS> START

.. Program: EVGCA, CIBCA T1015 Repair Level Diagnostic Part I, revision 2.0, 13
tests,
   at 14:16:01.92.
Testing: _PAA0

Test 1: Device Type/BIIC Configuration Register Test
Port Revision(Hex):      03

Link Revision(Hex):      43

Bca Adapter SelfTest (Port & Link) Completed Successfully

Test 2: BI Control and Status Register Test
Test 3: BI Required Registers Test
 Bca Register Access Test
 Bca Register Data Pattern Test for Bus Error Register
 Bca Register Data Pattern Test for Error Interrupt Control Register
 Bca Register Data Pattern Test for Interrupt Destination Register
 Bca Register Data Pattern Test for Ip Interrupt Mask Register
 Bca Register Data Pattern Test for Ip Interrupt Destination Register
 Bca Register Data Pattern Test for Ip Interrupt Source Register
Test 4: General Purpose Device Registers Test
 Bca Register Data Pattern Test for Port Queue Block Base Register
 Bca Register Data Pattern Test for Port Failing Address Register
 Bca Register Data Pattern Test for Port Parameter Register
 Bca Register Data Pattern Test for Port Error Register
Test 5: User CSR Space Register Test
 Bca Register Data Pattern Test for Receive Console Data Register
Test 6: PSR/PMCSR Register Test
Test 7: CIBCA Specific Register Test
 Bca Register Data Pattern Test for the Port Maintenance Control/Status Register
Test 8: Local Store Address Read/Write Test
Test 9: Local Store Data Read/Write Test
Test 10: Local Store Dynamic Memory Test
Test 11: Control Store (CS) Address Test
Test 12: Control Store Read/Write Ram Test
Test 13: Control Store Ram Dynamic Memory Test
.. End of run, 0 errors detected, pass count is 1,
   time is 28-DEC-1987 14:17:29.44
```

Trace Printout for Repair Diagnostic EVGCA

Repair-Level Diagnostics for the CIBCA (cont)

- o EVGCA
  - first of five repair-level diagnostics for CIBCA
  - Thirteen tests
- o Section

REGISTER	BCA Register stuck bit tests

- o Event Flags
  - Flag 1 enables BI selftest for each pass

```
DS> RUN EVGCB

.. Program: EVGCB, CIBCA T1015 Repair Level Diagnostic Part II, revision 2.1, 16
 tests,
   at 14:17:54.30.
Testing: _PAA0

Test 1: <EEPROM Integrity Verification Test>
Test 2: <Internal Bus Branch/Sequencer Jump Test - (Ucode Used)>
Test 3: <AM2910-A Microprogram Controller UPC+1 Test>
Test 4: <Micro-Controller Call To/ Return From Subroutine Test>
Test 5: <AM2910-A Micro-controller Pop Micro Stack Test (Ucode Used)>
Test 6: <AMD29116 Single Operand Instruction Test (Src) to (Dst)>
Test 7: <29116 Double Operand Instruction Test (Src) to (Dst)>
Test 8: <Instruction Tests for 29116 Shift Left and Right.>
Test 9: <Uproc Instr Test for Rotate By n Bit(s) Shift Instructions>
Test 10: <29116 Uproc Instr Tst for Bit Oriented Instr.>
Test 11: <Uproc Bit Oriented Instr RAM/ACC Minus Bit "N".>
Test 12: <Bit Oriented Instr Set Dlatch Bit N, Dlatch(Src) & Ram(Dst)>
Test 13: <Bit Oriented Instr, Reset BIT in RAM Test.>
Test 14: <Bit Oriented Instr, Reset BIT in Accumulator Test.>
Test 15: <Bit Oriented Instr Reset BIT in DLATCH Test.>
Test 16: <BIT ORIENTED AND BRANCH INSTR RAM Bit Test>
.. End of run, 0 errors detected, pass count is 1,
   time is 28-DEC-1987 14:18:40.56
```

Trace Printout for Repair Diagnostic EVGCB

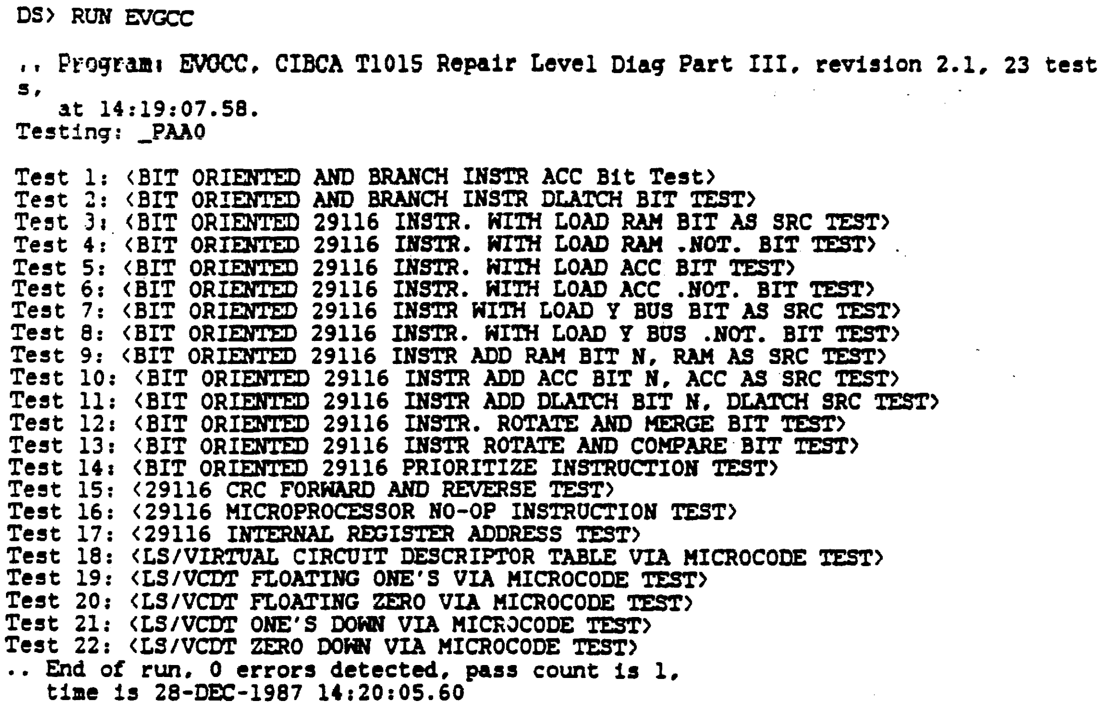
Repair-Level Diagnostics for the CIBCA (cont)

- o EVGCB
  - second of five repair-level diagnostics for CIBCA
  - Sixteen tests

- o Section

REGISTER	Checks for correct EEPROM data
MICROCONTROLLER	Microprogram control functions
MICROPROCESSOR	Check 29116

- o Event Flags
  - Flag 1 enables BI selftest for each pass
  - Flag 6 used for microlooping



```
DS> RUN EVGCC

.. Program: EVGCC, CIBCA T1015 Repair Level Diag Part III, revision 2.1, 23 test
s,
   at 14:19:07.58.
Testing: _PAA0

Test 1: <BIT ORIENTED AND BRANCH INSTR ACC Bit Test>
Test 2: <BIT ORIENTED AND BRANCH INSTR DLATCH BIT TEST>
Test 3: <BIT ORIENTED 29116 INSTR. WITH LOAD RAM BIT AS SRC TEST>
Test 4: <BIT ORIENTED 29116 INSTR. WITH LOAD RAM .NOT. BIT TEST>
Test 5: <BIT ORIENTED 29116 INSTR. WITH LOAD ACC BIT TEST>
Test 6: <BIT ORIENTED 29116 INSTR. WITH LOAD ACC .NOT. BIT TEST>
Test 7: <BIT ORIENTED 29116 INSTR WITH LOAD Y BUS BIT AS SRC TEST>
Test 8: <BIT ORIENTED 29116 INSTR. WITH LOAD Y BUS .NOT. BIT TEST>
Test 9: <BIT ORIENTED 29116 INSTR ADD RAM BIT N, RAM AS SRC TEST>
Test 10: <BIT ORIENTED 29116 INSTR ADD ACC BIT N, ACC AS SRC TEST>
Test 11: <BIT ORIENTED 29116 INSTR ADD DLATCH BIT N, DLATCH SRC TEST>
Test 12: <BIT ORIENTED 29116 INSTR. ROTATE AND MERGE BIT TEST>
Test 13: <BIT ORIENTED 29116 INSTR ROTATE AND COMPARE BIT TEST>
Test 14: <BIT ORIENTED 29116 PRIORITIZE INSTRUCTION TEST>
Test 15: <29116 CRC FORWARD AND REVERSE TEST>
Test 16: <29116 MICROPROCESSOR NO-OP INSTRUCTION TEST>
Test 17: <29116 INTERNAL REGISTER ADDRESS TEST>
Test 18: <LS/VIRTUAL CIRCUIT DESCRIPTOR TABLE VIA MICROCODE TEST>
Test 19: <LS/VCDT FLOATING ONE'S VIA MICROCODE TEST>
Test 20: <LS/VCDT FLOATING ZERO VIA MICROCODE TEST>
Test 21: <LS/VCDT ONE'S DOWN VIA MICROCODE TEST>
Test 22: <LS/VCDT ZERO DOWN VIA MICROCODE TEST>
.. End of run, 0 errors detected, pass count is 1,
   time is 28-DEC-1987 14:20:05.60
```

Trace Printout for Repair Diagnostic EVGCC

Repair-Level Diagnostics for the CIBCA (cont)

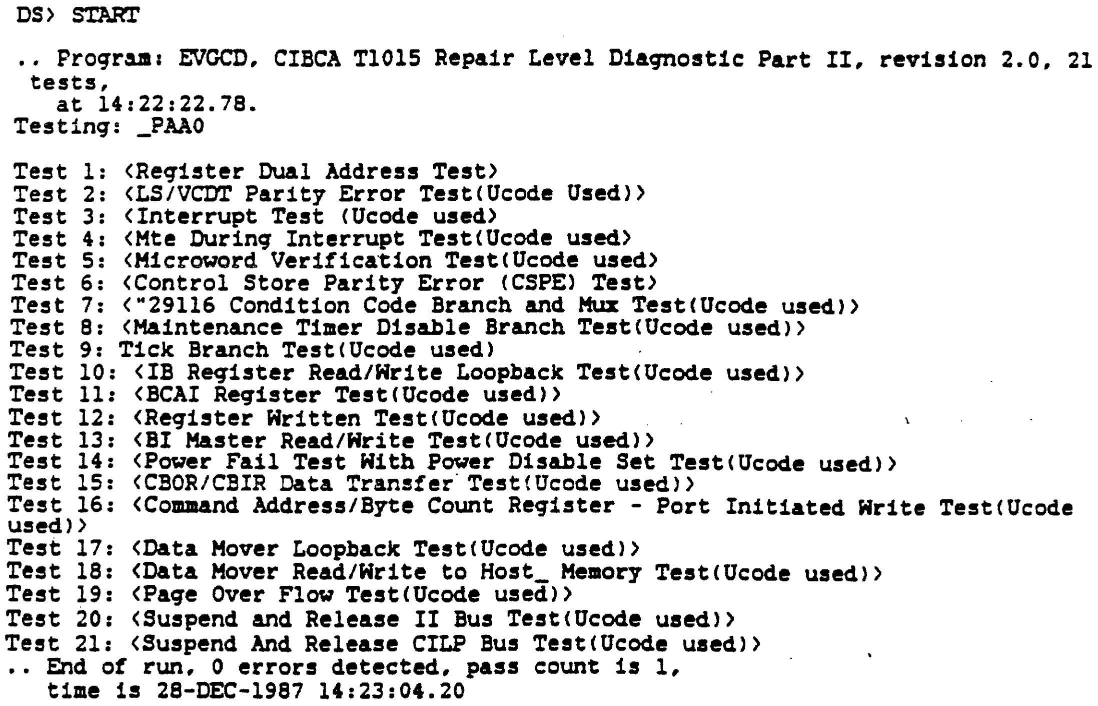
- o EVGCC
  - third of five repair-level diagnostics for CIBCA.
  - twenty-three tests.

- o Section

MICROPROCESSOR	Check 29116
LOCAL_STORE	Test for stuck bits in local store and VCDT

- o Event Flags
  - Flag 1 enables BI selftest for each pass.
  - Flag 6 used for microlooping. MANUF. ONLY

Repair-Level Diagnostics for the CIBCA (cont)



```
DS> START

.. Program: EVGCD, CIBCA T1015 Repair Level Diagnostic Part II, revision 2.0, 21
 tests,
   at 14:22:22.78.
Testing: _PAA0

Test 1: <Register Dual Address Test>
Test 2: <LS/VCDT Parity Error Test(Ucode Used)>
Test 3: <Interrupt Test (Ucode used>
Test 4: <Mte During Interrupt Test(Ucode used>
Test 5: <Microword Verification Test(Ucode used>
Test 6: <Control Store Parity Error (CSPE) Test>
Test 7: <"29116 Condition Code Branch and Mux Test(Ucode used)>
Test 8: <Maintenance Timer Disable Branch Test(Ucode used)>
Test 9: Tick Branch Test(Ucode used)
Test 10: <IB Register Read/Write Loopback Test(Ucode used)>
Test 11: <BCAI Register Test(Ucode used)>
Test 12: <Register Written Test(Ucode used)>
Test 13: <BI Master Read/Write Test(Ucode used)>
Test 14: <Power Fail Test With Power Disable Set Test(Ucode used)>
Test 15: <CBOR/CBIR Data Transfer Test(Ucode used)>
Test 16: <Command Address/Byte Count Register - Port Initiated Write Test(Ucode
used)>
Test 17: <Data Mover Loopback Test(Ucode used)>
Test 18: <Data Mover Read/Write to Host_ Memory Test(Ucode used)>
Test 19: <Page Over Flow Test(Ucode used)>
Test 20: <Suspend and Release II Bus Test(Ucode used)>
Test 21: <Suspend And Release CILP Bus Test(Ucode used)>
.. End of run, 0 errors detected, pass count is 1,
   time is 28-DEC-1987 14:23:04.20
```

Trace Printout for Repair Diagnostic EVGCD

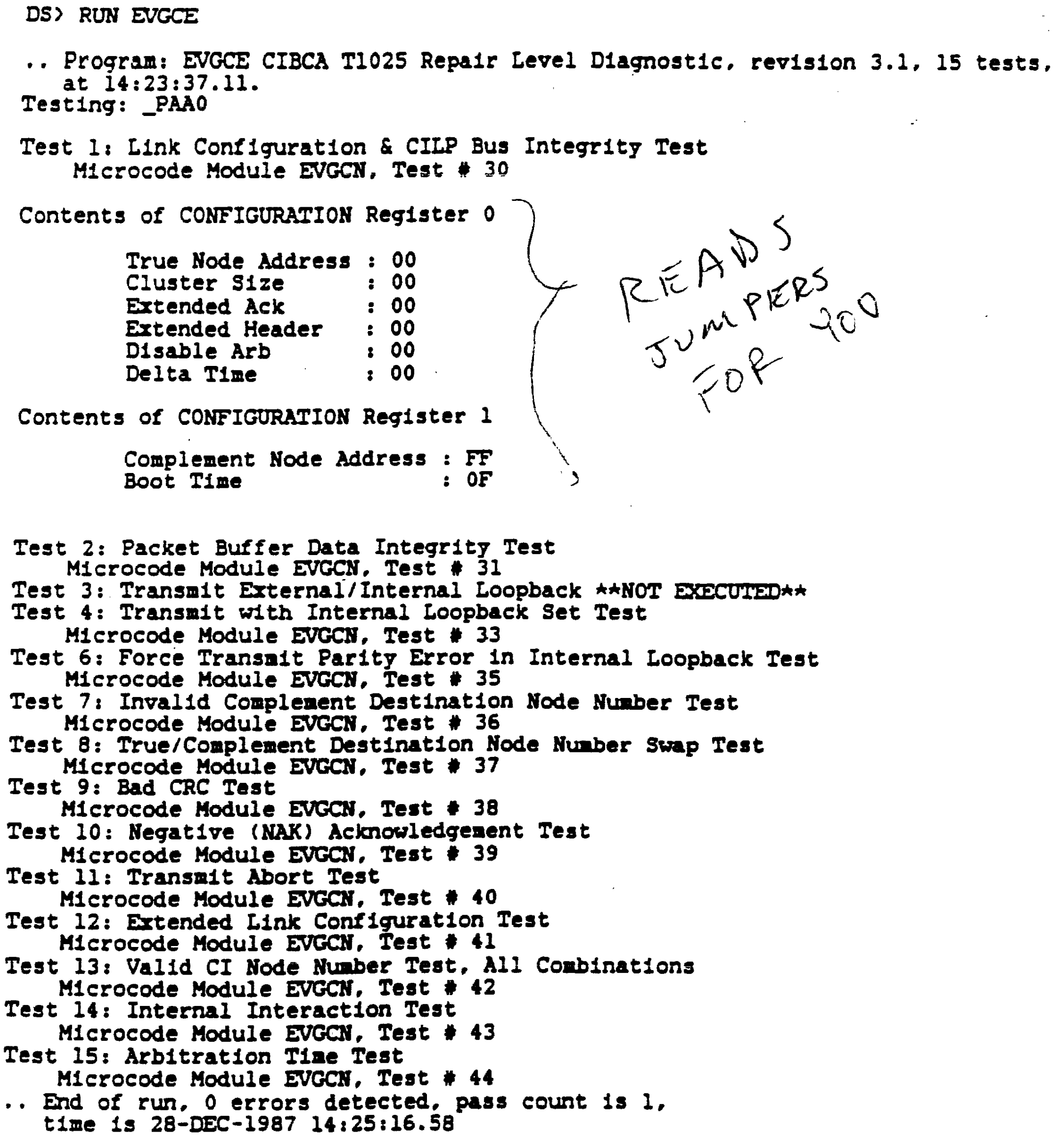
Repair-Level Diagnostics for the CIBCA (cont)

- o EVGCD
  - fourth of five repair-level diagnostics for CIBCA.
  - twenty-one tests.

- o Section

REGISTER	Various registers checked
MICROCONTROLLER	Microword verification
MICROPROCESSOR	Condition code branch check
INTERRUPT	MTE and parity error interrupt check
DATAMOVER	BI-to-CI data flow check
LINK	CBOR/CBIR data transfer test

- o Event Flags
  - none available



```
DS> RUN EVGCE

.. Program: EVGCE CIBCA T1025 Repair Level Diagnostic, revision 3.1, 15 tests,
   at 14:23:37.11.
Testing: _PAA0

Test 1: Link Configuration & CILP Bus Integrity Test
    Microcode Module EVGCN, Test # 30

Contents of CONFIGURATION Register 0

        True Node Address : 00
        Cluster Size      : 00
        Extended Ack      : 00
        Extended Header   : 00
        Disable Arb       : 00
        Delta Time        : 00

Contents of CONFIGURATION Register 1

        Complement Node Address : FF
        Boot Time               : 0F


Test 2: Packet Buffer Data Integrity Test
    Microcode Module EVGCN, Test # 31
Test 3: Transmit External/Internal Loopback **NOT EXECUTED**
Test 4: Transmit with Internal Loopback Set Test
    Microcode Module EVGCN, Test # 33
Test 6: Force Transmit Parity Error in Internal Loopback Test
    Microcode Module EVGCN, Test # 35
Test 7: Invalid Complement Destination Node Number Test
    Microcode Module EVGCN, Test # 36
Test 8: True/Complement Destination Node Number Swap Test
    Microcode Module EVGCN, Test # 37
Test 9: Bad CRC Test
    Microcode Module EVGCN, Test # 38
Test 10: Negative (NAK) Acknowledgement Test
    Microcode Module EVGCN, Test # 39
Test 11: Transmit Abort Test
    Microcode Module EVGCN, Test # 40
Test 12: Extended Link Configuration Test
    Microcode Module EVGCN, Test # 41
Test 13: Valid CI Node Number Test, All Combinations
    Microcode Module EVGCN, Test # 42
Test 14: Internal Interaction Test
    Microcode Module EVGCN, Test # 43
Test 15: Arbitration Time Test
    Microcode Module EVGCN, Test # 44
.. End of run, 0 errors detected, pass count is 1,
   time is 28-DEC-1987 14:25:16.58
```

READS JUMPERS FOR 400

Trace Printout for Repair Diagnostic EVGCE

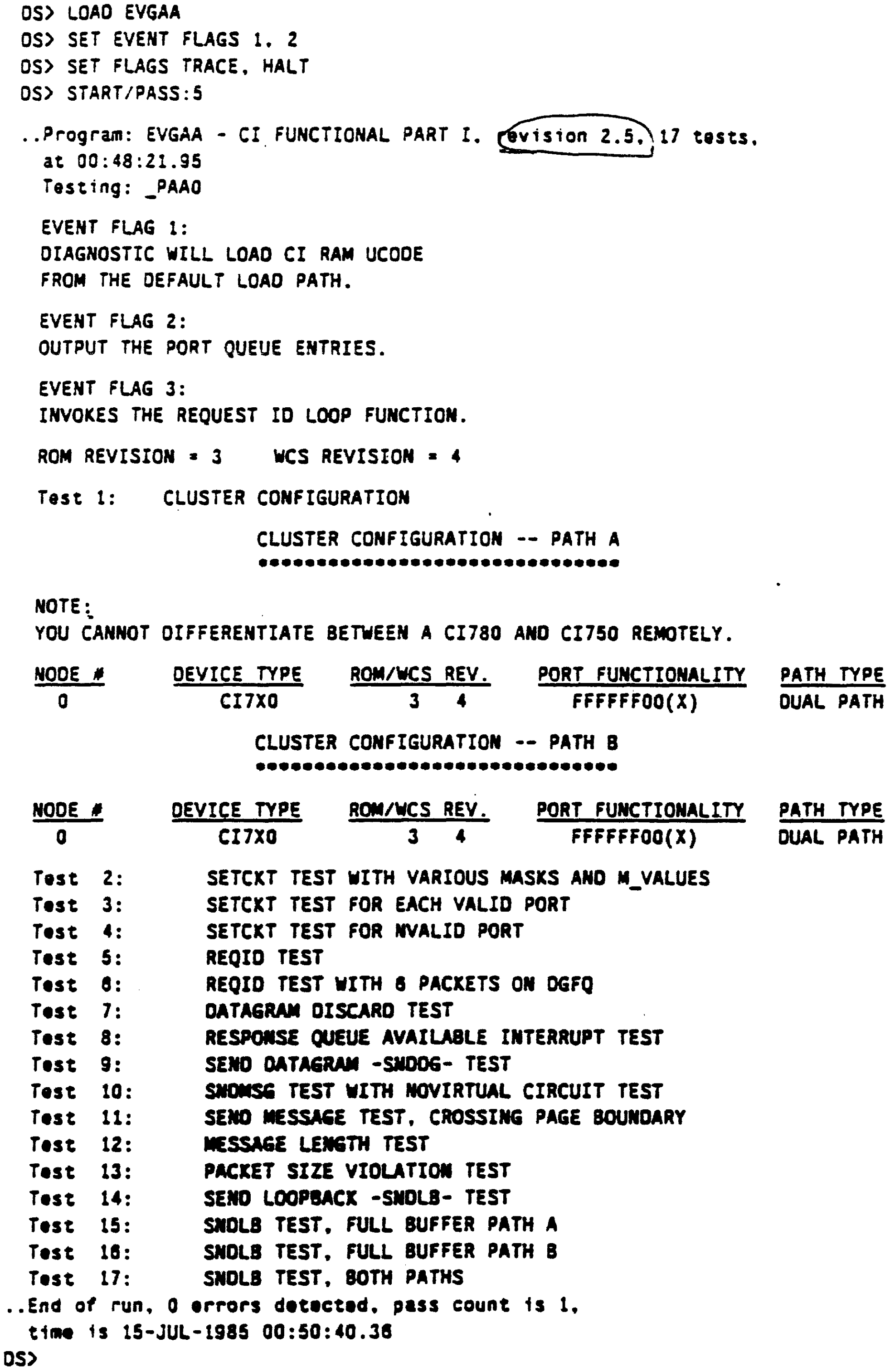
Repair-Level Diagnostics for the CIBCA (cont)

- EVGCE
  - fifth of five repair-level diagnostics for CIBCA.
  - fifteen tests.
- Section

EXTERNAL_LOOP	Loop externally on CI

- Event Flags
  - none available

Functional Diagnostics



```
DS> LOAD EVGAA
DS> SET EVENT FLAGS 1, 2
DS> SET FLAGS TRACE, HALT
DS> START/PASS:5

..Program: EVGAA - CI FUNCTIONAL PART I, revision 2.5, 17 tests,
  at 00:48:21.95
  Testing: _PAA0

  EVENT FLAG 1:
  DIAGNOSTIC WILL LOAD CI RAM UCODE
  FROM THE DEFAULT LOAD PATH.

  EVENT FLAG 2:
  OUTPUT THE PORT QUEUE ENTRIES.

  EVENT FLAG 3:
  INVOKES THE REQUEST ID LOOP FUNCTION.

  ROM REVISION = 3     WCS REVISION = 4

  Test 1:    CLUSTER CONFIGURATION

                       CLUSTER CONFIGURATION -- PATH A
                       ******************************

  NOTE:
  YOU CANNOT DIFFERENTIATE BETWEEN A CI780 AND CI750 REMOTELY.

  NODE #      DEVICE TYPE    ROM/WCS REV.    PORT FUNCTIONALITY    PATH TYPE
    0            CI7X0          3    4          FFFFFF00(X)        DUAL PATH

                       CLUSTER CONFIGURATION -- PATH B
                       ******************************

  NODE #      DEVICE TYPE    ROM/WCS REV.    PORT FUNCTIONALITY    PATH TYPE
    0            CI7X0          3    4          FFFFFF00(X)        DUAL PATH

  Test  2:        SETCKT TEST WITH VARIOUS MASKS AND M_VALUES
  Test  3:        SETCKT TEST FOR EACH VALID PORT
  Test  4:        SETCKT TEST FOR NVALID PORT
  Test  5:        REQID TEST
  Test  6:        REQID TEST WITH 6 PACKETS ON DGFQ
  Test  7:        DATAGRAM DISCARD TEST
  Test  8:        RESPONSE QUEUE AVAILABLE INTERRUPT TEST
  Test  9:        SEND DATAGRAM -SNDDG- TEST
  Test  10:       SNDMSG TEST WITH NOVIRTUAL CIRCUIT TEST
  Test  11:       SEND MESSAGE TEST, CROSSING PAGE BOUNDARY
  Test  12:       MESSAGE LENGTH TEST
  Test  13:       PACKET SIZE VIOLATION TEST
  Test  14:       SEND LOOPBACK -SNDLB- TEST
  Test  15:       SNDLB TEST, FULL BUFFER PATH A
  Test  16:       SNDLB TEST, FULL BUFFER PATH B
  Test  17:       SNDLB TEST, BOTH PATHS
..End of run, 0 errors detected, pass count is 1,
  time is 15-JUL-1985 00:50:40.36
DS>
```

Trace Printout for Functional Diagnostic EVGAA

Functional Diagnostics (Cont.)

- EVGAA
  - First of two CI functional diagnostics for the CI780, CI750, or CIBCI.
  - Seventeen tests.
- Event Flags
  - Flag 1 is used to load CI microcode before the start of each pass. If clear, the code presently in CI RAM will be used.
  - Flag 2 outputs the contents of the port queue entries. ← PRINTS BUNCH OF JUNK DONT NEED TO SEE NO GOOD DATA!
  - Flag 3 is used for monitoring a particular node and path.

    Set event flg 3 TO TARGET TESTS ON ANOTHER PARTICULAR NODE — LIKE AN HSC!
- Operation
  - QUICK feature not implemented.
  - No Summary report is issued.
  - Execution time is between 40 to 60 seconds (depending on the particular interface being used).

NOTE: When using Event Flag 1 (loading new microcode before starting functional diagnostics), be sure CI780.BIN is accessible via the DEFAULT LOAD device. OR CIBCA.BIN

Functional Diagnostics (Cont.)

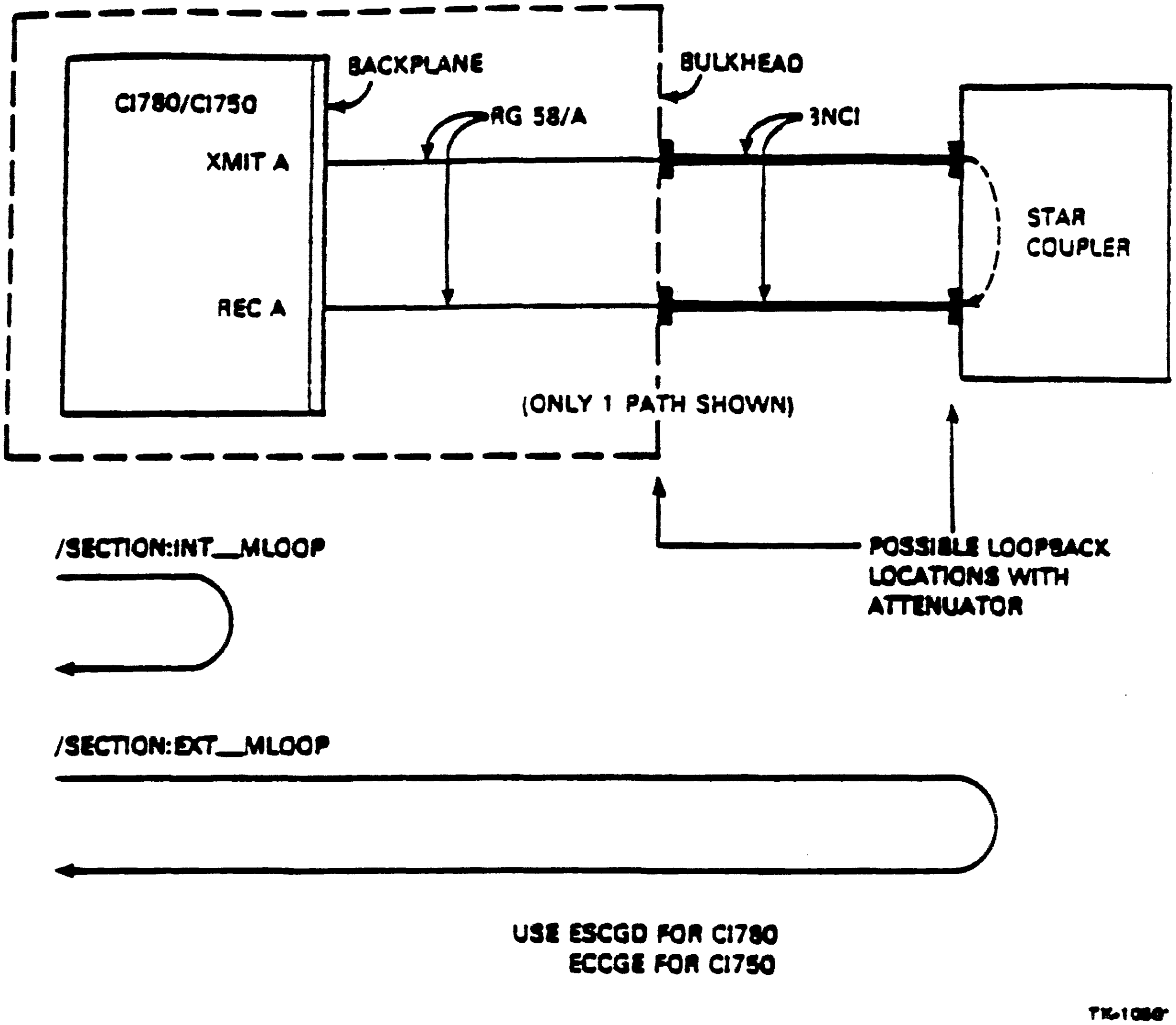
```
DS> LOAD EVGAB
DS> CLEAR EVENT FLAG 1, 2
DS> SET FLAGS TRACE, HALT
DS> START/PASS:5

..Program: EVGAB - CI FUNCTIONAL PART II, revision 2.5, 12 tests,
  at 00:50:54.31.
  Testing: _PAA0

  ROM REVISION = 3    WCS REVISION = 4

  Test  1:    SEND DATA TEST, WITH OFFSET COMBINATIONS
  Test  2:    REQUEST DATA TEST, WITH OFFSET COMBINATIONS
  Test  3:    INVALIDATE TRANSLATION CACHE TEST
  Test  4:    SNDMDAT TEST, ENABLED/MAINTENANCE STATE
  Test  5:    SNDMDAT TEST, ENABLED STATE
  Test  6:    REQMDAT TEST, ENABLED/MAINT STATE
  Test  7:    REQMDAT TEST, ENABLED STATE
  Test  8:    SEND RESET TEST IN ENABLED STATE
  Test  9:    QUEUE CONTENTION TEST
  Test  10:   BUFFER READ ACCESS TEST
  Test  11:   BUFFER WRITE ACCESS TEST
  Test  12:   WRITE TO GLOBAL BUFFER TEST
..End of run, 0 errors detected, pass count is 1,
  time is 15-JUL-1985 00:52:29.92
DS>
```

Trace Printout for Functional Diagnostic EVGAB

Functional Diagnostics (Cont.)

- EVGAB
  - Second of two CI functional diagnostics for the CI780, CI750, or CIBCI.
  - Twelve tests.
- Event Flags
  - Flag 1 is used to load CI microcode before the start of each pass. If clear, the code presently in CI RAM will be used.
  - Flag 2 outputs the contents of the port queue entries.
- Operation
  - QUICK feature is not implemented.
  - No Summary report is issued.
  - Execution time is between 40 to 60 seconds (depending on the particular interface being used).

NOTE: When using Event Flag 1 (loading new microcode before starting functional diagnostics), be sure CI780.BIN is accessible via the LOAD PATH you are presently using.



Looping Data on the CI Bus

VMS V5

VAXPAX ~~V33~~ BAD – ONLINE DIAG DRIVERS SCREWED UP

V34 GOOD

EVXCI: Cluster Exerciser

- A level 2R multi-purpose exerciser that provides local CI interface functional testing, as well as a means of determining the ability of VAXcluster nodes, to reliably communicate using the CI Bus.

- The exerciser is found on the VAXPAX distribution medium under the product name of CIE081. NOT IN REV 33 DIAGS GO TO REV 34

- CIE081 must be installed under SYS$UPDATE:VMSINSTAL into the SYS$SYSMAINTENANCE directory.

- Follow these steps to run the CI Exerciser:
  - Install CIE081 from the VAXPAX kit.
  - Execute CIELOAD.COM at EACH node.
  - Define CIE as a foreign command.
  - Run the exerciser by typing CIE.
  - On nodes that the exerciser cannot find responders on, execute the CIELOAD.COM to load the drivers.
  - Run the exerciser again.

$> TYPE EUNDX, SEQ
WILL GIVE REV OF VAXPAX THAT DIAGS CAME FROM

ON HSC'S

```
    HSC> RUN SETSHO ↵
    SETSHO> SHOW LOAD ↵ —
             LOOK FOR RS. IF NOT THERE DO
(1) SETSHO> SET LOAD RS ↵
(2) SETSHO> ENABLE REBOOT ↵
(3) SETSHO> EXIT ↵
        (4) REBOOT? YES ↵
```

Sample Run of EVXCI

```
                         Welcome to node WRAITH on the

                  Santa Clara Educational Services Hardware Cluster

Please enter your username followed by a carriage return. Now, enter your password (it will not
echo) and a carriage return.

If you have problems logging on, see your instructor!


+--------------------------------------------------------------+
|        THIS IS THE STANDARD CLASS VAX/VMS SYSTEM! ! ! !      |
+--------------------------------------------------------------+


Username:  FIELD
Password:
          Welcome to VAX/VMS version V4.4 on node WRAITH
   Last interactive login on Friday, 11-MAR-1988 09:04
   Last non-interactive login on Friday, 19-FEB-1988 15:58
$  SET DEFAULT SYS$MAINTENANCE
$  DIR

Directory SYS$SYSROOT:[SYSMAINT]

DUCT.EXE;1          DUCT.RELEASE_NOTES;1
DUCT_CURR_ACCT.TMP;1

Total of 3 files.

Directory SYS$COMMON:[SYSMAINT]

8200_RELEASE_NOTES.DOC;1                ACCEPT.COM;1        ACCESS.BPN_8600;1
ANYBOO.COM_8600;1   BINCOM.SAV_8600;1   BOOT.HLP_8600;1     BOOT._8600;1
CCSNOOP.BIN;1       CDF860.DAT_8600;2   CDF865.DAT_8600;2   CI780.BIN;1
CIBCA.BIN;14        CIBOO.COM_8600;1    CIE081.A;3          CIE081.B;2
CIE081.C;1          CIE081_RELEASE_NOTES.DOC;1              CIELOAD.COM;1
CNSL.COM_8600;1     CONLIST.LIS_8600;40 CONLIST_NODIAG.LIS_8600;22
^Y
```

$> DIR CIE*.*

Sample Run of EVXCI (Cont.)

```
$ @SYS$UPDATE:VMSINSTAL

    VAX/VMS Software Product Installation Procedure V4.4

It is 13-MAR-1988 at 10:48.
Enter a question mark (?) at any time for help.
%VMSINSTAL-W-NOTSYSTEM, You are not logged in to the SYSTEM account.
%VMSINSTAL-W-DECNET, Your DECnet network is up and running.
* Do you want to continue anyway [NO]?  YES
* Are you satisfied with the backup of your system disk [YES]?  YES
* Where will the distribution volumes be mounted: SYS$MAINTENANCE:

Enter the products to be processed from the first distribution volume set.
* Products: CIE081
The following products will be processed:
  CIE V8.1
        Beginning installation of CIE V8.1 at 10:48
%VMSINSTAL-I-RESTORE, Restoring product saveset A...
%VMSINSTAL-I-RESTORE, Restoring product saveset B...
%VMSINSTAL-I-RESTORE, Restoring product saveset C...

        Linking drivers and EVXCI...

Enter Y[es] to purge old files: YES

        Control will now be returned to VMSINSTAL, which will actually
        move the files to their destination directories.

CIE version 8.1 has been copied to SYS$COMMON:[SYSMAINT]
```

To run the CI Exerciser, make CIE a foreign command in your LOGIN.COM file.

Example:

```
       CIE :== " $ SYS$COMMON:[SYSMAINT]Evxci.exe "
```

```
%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...
       Installation of CIE V8.1 completed at 10:49

Enter the products to be processed from the next distribution volume set.
* Products: EXIT
       VMSINSTAL procedure done at 10:50
```

Sample Run of EVXCI (Cont.)

```
$ @CIELOAD

 CXDRIVER and CYDRIVER have been loaded successfully.

$ CIE == "$ SYS$COMMON:[SYSMAINT]EVXCI.EXE"
$ CIE

Local Port can NOT connect with remote node, Node will be deselected.
       Please, make sure that ULAIRI has loaded responder driver (CYDRIVER)
       Node ULAIRI has been deselected.

----------------------------------------------------------------------
  Testing from WRAITH,   PAA0 Node address 02,   Number of nodes = 02
**********************************************************************

       Node Name     Node #        Hardware Type      Status

       SAURON         01             HS50              Open
       MELKOR         00             HS50              Open

...  Computer Interconnect Exerciser, ZZ-EVXCI Version 8.1
       Started at 13-MAR-1988 10:55:12.11

Testing WRAITH, node number 02

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
TEST #15 CONTROLLER WRITE BUFFER
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
       Test skipped, no third party at this time.
```

Sample Run of EVXCI (Cont.)

```
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
       Test skipped, no third party at this time.
TEST #20 ACTIVITY GENERATION
       Test skipped, no third party at this time.
TEST #21 PERFORMANCE COUNTERS
       Test skipped, no third party at this time.

Testing SAURON, node number 01

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
       Test skipped, unit does not support required CI functions.
TEST #15 CONTROLLER WRITE BUFFER
       Test skipped, unit does not support required CI functions.
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
TEST #20 ACTIVITY GENERATION
TEST #21 PERFORMANCE COUNTERS
NODE       P0ACK      P0NAK      P0NORSP     P1ACK      P1NAK       P1NORSP
 02      00000020   00000000   00000000   00000020   00000000   00000000
 01      00000022   00000000   00000000   0000001E   00000000   00000000

Testing MELKOR, node number 00

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
```

Sample Run of EVXCI (Cont.)

```
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
        Test skipped, unit does not support required CI functions.
TEST #15 CONTROLLER WRITE BUFFER
        Test skipped, unit does not support required CI functions.
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
TEST #20 ACTIVITY GENERATION
TEST #21 PERFORMANCE COUNTERS
NODE      P0ACK      P0NAK       P0NORSP     P1ACK       P1NAK       P1NORSP
 02     00000037   00000000    00000000    0000002B    00000000    00000000
 00     0000001F   00000000    00000000    00000021    00000000    00000000
 01     0000000F   00000000    00000000    00000013    00000000    00000000
...  End of run, 3 nodes tested, 0 errors detected, pass count is 1,
        Ending time 13-MAR-1988 10:56:26.16
$ SET HOST ULAIRI


                              Welcome to node ULAIRI on the
                     Santa Clara Educational Services Hardware Cluster

Please enter your username followed by a carriage return.
Now, enter your password (it will not echo) and a carriage
return.

If you have problems logging on, see your instructor!

+--------------------------------------------------------------+
|                                                              |
|      THIS IS THE STANDARD CLASS VAX/VMS SYSTEM! ! ! !        |
|                                                              |
+--------------------------------------------------------------+


Username: FIELD
Password:
        Welcome to VAX/VMS version V4.4 on node ULAIRI
   Last interactive login on Sunday, 13-MAR-1988 11:18
   Last non-interactive login on Friday, 19-FEB-1988 15:58
```

Sample Run of EVXCI (Cont.)

```
$ @CIELOAD

 CXDRIVER and CYDRIVER have been loaded successfully.
$   LO
    FIELD      logged out at 13-MAR-1988 11:31:48.43
%REM-S-END, control returned to node _WRAITH::
$ CIE == "$ SYS$COMMON:[SYSMAINT]EVXCI.EXE"
$ CIE
------------------------------------------------------------------------------
  ATesting from WRAITH, PAA0 Node address 02, Number of nodes = 03
******************************************************************************
       Node Name       Node #         Hardware       Type Status

       SAURON           01             HS50           Open
       MELKOR           00             HS50           Open
       ULAIRI           04             8600           Open
...  Computer Interconnect Exerciser, ZZ-EVXCI Version 8.1
         Started at 13-MAR-1988 11:19:35.23

Testing WRAITH, node number 02

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
TEST #15 CONTROLLER WRITE BUFFER
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
       Test skipped, no third party at this time.
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
       Test skipped, no third party at this time.
TEST #20 ACTIVITY GENERATION
       Test skipped, no third party at this time.
TEST #21 PERFORMANCE COUNTERS
       Test skipped, no third party at this time.
```

Sample Run of EVXCI (Cont.)

```
Testing SAURON, node number 01

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
       Test skipped, unit does not support required CI functions.
TEST #15 CONTROLLER WRITE BUFFER
       Test skipped, unit does not support required CI functions.
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
TEST #20 ACTIVITY GENERATION
BTEST #21 PERFORMANCE COUNTERS
NODE      P0ACK      P0NAK     P0NORSP     P1ACK      P1NAK       P1NORSP
 02     00000024   00000000   00000000   0000001C   00000000    00000000
 01     0000001A   00000000   00000000   00000026   00000000    00000000

Testing MELKOR, node number 00

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
```

Sample Run of EVXCI (Cont.)

```
TEST #14 CONTROLLER READ BUFFER
       Test skipped, unit does not support required CI functions.
TEST #15 CONTROLLER WRITE BUFFER
       Test skipped, unit does not support required CI functions.
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
TEST #20 ACTIVITY GENERATION
TEST #21 PERFORMANCE COUNTERS
NODE      P0ACK     P0NAK      P0NORSP     P1ACK      P1NAK       P1NORSP
 02     00000035  00000000    00000000   0000002D   00000000    00000000
 00     0000001F  00000000    00000000   00000021   00000000    00000000
 01     00000013  00000000    00000000   0000000F   00000000    00000000

Testing ULAIRI, node number 04

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
TEST #15 CONTROLLER WRITE BUFFER
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
TEST #20 ACTIVITY GENERATION
TEST #21 PERFORMANCE COUNTERS
NODE      P0ACK     P0NAK      P0NORSP     P1ACK      P1NAK       P1NORSP
 02     00000C8B  00000000    00000300   00000CA0   00000000    00000000
 04     00000CA2  00000004    00000002   00000C3F   00000006    00000000
 01     00000013  00000000    00000001   00000015   00000000    00000002
 00     00000012  00000001    00000001   00000011   00000000    00000003
...  End of run, 4 nodes tested, 0 errors detected, pass count is 1,
       Ending time 13-MAR-1988 11:21:21.53
```

Sample Run of EVXCI (Cont.)

```
$ SET HOST ULAIRI

                                        Welcome to node ULAIRI on the
                                Santa Clara Educational Services Hardware Cluster

   Please enter your username followed by a carriage return.
   Now, enter your password (it will not echo) and a carriage
   return.

   If you have problems logging on, see your instructor!
```

```
         THIS IS THE STANDARD CLASS VAX/VMS SYSTEM! ! ! !
```

```
Username: FIELD
Password:
        Welcome to VAX/VMS version V4.4 on node ULAIRI
    Last interactive login on Sunday, 13-MAR-1988 11:18
    Last non-interactive login on Friday, 19-FEB-1988 15:58
$ CIE == "$ SYS$COMMON:[SYSMAINT]EVXCI.EXE"
$ CIE
------------------------------------------------------------------------------
Testing from ULAIRI, PAA0 Node address 04, Number of nodes = 03
******************************************************************************

       Node Name     Node #       Hardware Type      Status

       SAURON         01             HS50             Open
       WRAITH         02             8600             Open
       MELKOR         00             HS50             Open
...  Computer Interconnect Exerciser, ZZ-EVXCI Version 8.1
        Started at 13-MAR-1988 11:22:44.89

Testing ULAIRI, node number 04

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
```

Sample Run of EVXCI (Cont.)

```
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
TEST #15 CONTROLLER WRITE BUFFER
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
       Test skipped, no third party at this time.
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
       Test skipped, no third party at this time.
TEST #20 ACTIVITY GENERATION
       Test skipped, no third party at this time.
TEST #21 PERFORMANCE COUNTERS
       Test skipped, no third party at this time.

Testing SAURON, node number 01

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
       Test skipped, unit does not support required CI functions.
TEST #15 CONTROLLER WRITE BUFFER
       Test skipped, unit does not support required CI functions.
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
TEST #20 ACTIVITY GENERATION
TEST #21 PERFORMANCE COUNTERS
NODE      P0ACK      P0NAK       P0NORSP      P1ACK       P1NAK        P1NORSP
 04     00000024   00000000    00000000    0000001C    00000000     00000000
 01     00000025   00000000    00000000    0000001B    00000000     00000000
```

Sample Run of EVXCI (Cont.)

```
Testing WRAITH, node number 02

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
TEST #15 CONTROLLER WRITE BUFFER
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
TEST #20 ACTIVITY GENERATION
TEST #21 PERFORMANCE COUNTERS
NODE       P0ACK       P0NAK       P0NORSP      P1ACK       P1NAK        P1NORSP
 04      00000E18    00000000    00000001     00000E95    00000004     00000300
 02      00000E29    00000005    00000000     00000E62    00000004     00000301
 01      00000011    00000000    00000002     0000001A    00000005     00000000

Testing MELKOR, node number 00

TEST #1  LOCAL CONFIGURATION
TEST #2  CONNECT
TEST #3  BASIC MESSAGE
TEST #4  MESSAGE DATA
TEST #5  MULTIPLE MESSAGE
TEST #6  CTP FUNCTIONALITY
TEST #7  REMOTE CONFIGURATION
TEST #8  SEND DATAGRAM
TEST #9  RECEIVE DATAGRAM
TEST #10 DATAGRAM DATA
TEST #11 MULTIPLE DATAGRAM
TEST #12 RESPONDER WRITE BUFFER
TEST #13 RESPONDER READ BUFFER
TEST #14 CONTROLLER READ BUFFER
        Test skipped, unit does not support required CI functions.
TEST #15 CONTROLLER WRITE BUFFER
        Test skipped, unit does not support required CI functions.
```

Sample Run of EVXCI (Cont.)

```
TEST #16 READ WRITE BUFFER DATA
TEST #17 MULTI READ WRITE BUFFER DATA
TEST #18 THIRD PARTY CONFIGURATION
TEST #19 THIRD PARTY READ WRITE BUFFER DATA
TEST #20 ACTIVITY GENERATION
TEST #21 PERFORMANCE COUNTERS
NODE      POCK       PONAK       PONORSP     P1ACK       P1NAK        P1NORSP
04     0000026C   00000000    00000000    000002C3    00000000     00000000
00     0000001C   00000000    00000000    00000025    00000003     00000001
01     00000012   00000001    00000001    00000011    00000000     00000000
02     000002DE   00000001    00000000    0000031E    00000004     00000301
...  End of run, 4 nodes tested, 0 errors detected, pass count is 1,
       Ending time 13-MAR-1988 11:24:34.16

$ LO
  FIELD        logged out at 13-MAR-1988 11:31:48.43
%REM-S-END, control returned to node _WRAITH::
$ LO
  FIELD        logged out at 13-MAR-1988 11:31:48.43
```

CI PORT REGISTERS

CI Port Registers

Lesson Introduction

The CI Port has two types of registers called hardware registers and microcode registers. The hardware registers are defined by the hardware at all times. The microcode registers are created in the Local Store by the microcode upon initialization and are defined only when the functional microcode is running.

This lesson addresses the location and contents of the CI Port registers.

Lesson Objectives

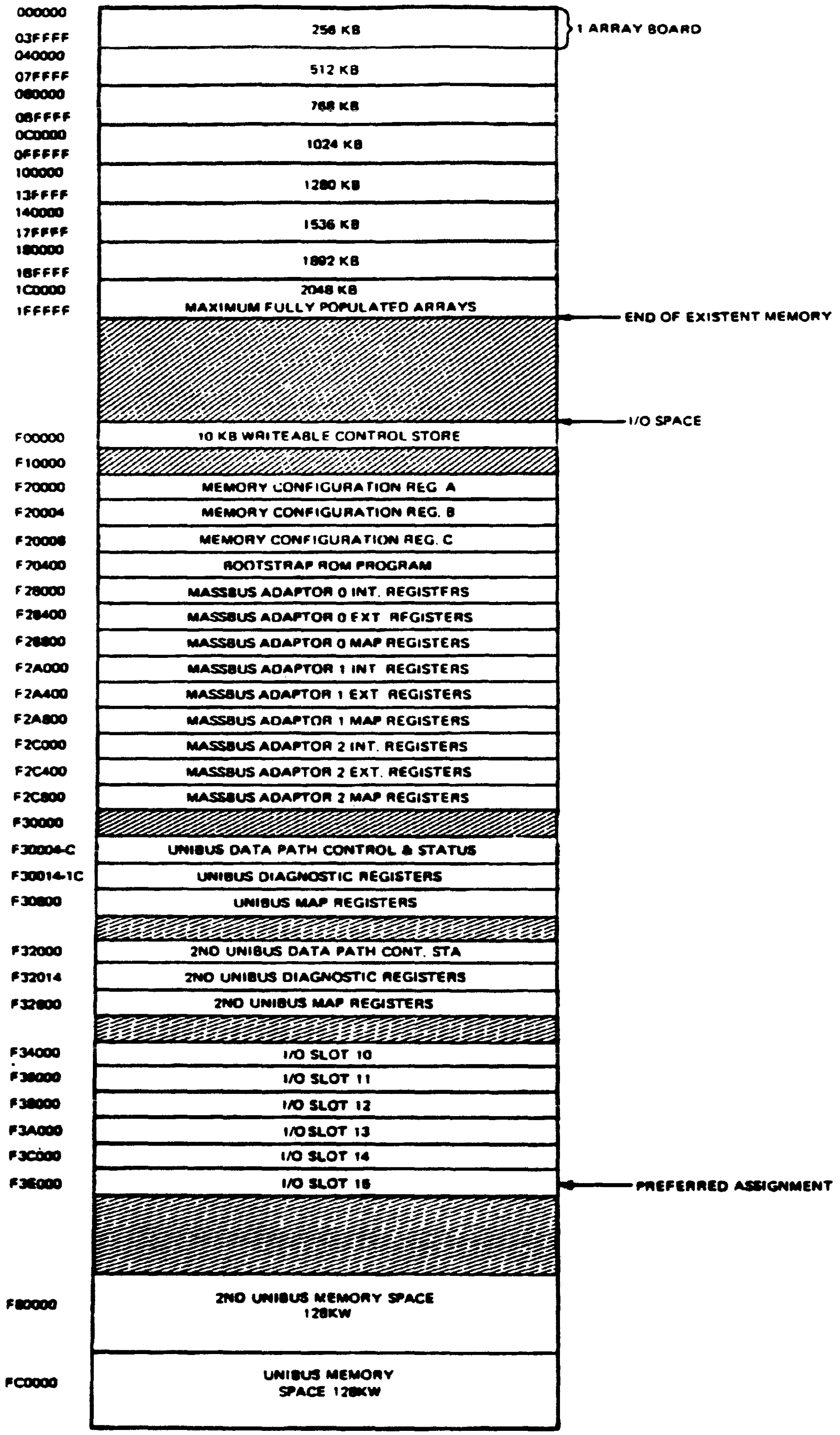
1. Analyze an event log printout to find a suspected defective CI FRU.
2. Calculate a registers address and then examine that register from the VAX Console Terminal.
3. Examine and deposit data to and from the Control Store, Local Store, and VCDT.
4. Identify the current condition of the CI Port from the interface registers.

Lesson Outline

I. Physical Address Space Map
II. CI Port Hardware Registers
III. CI Port Microcode Registers
IV. Device Specific Registers

CI750 Address Assignment

- Preferred assignment within VAX 750 address map is I/O slot 15.
- Base address is F3E000.

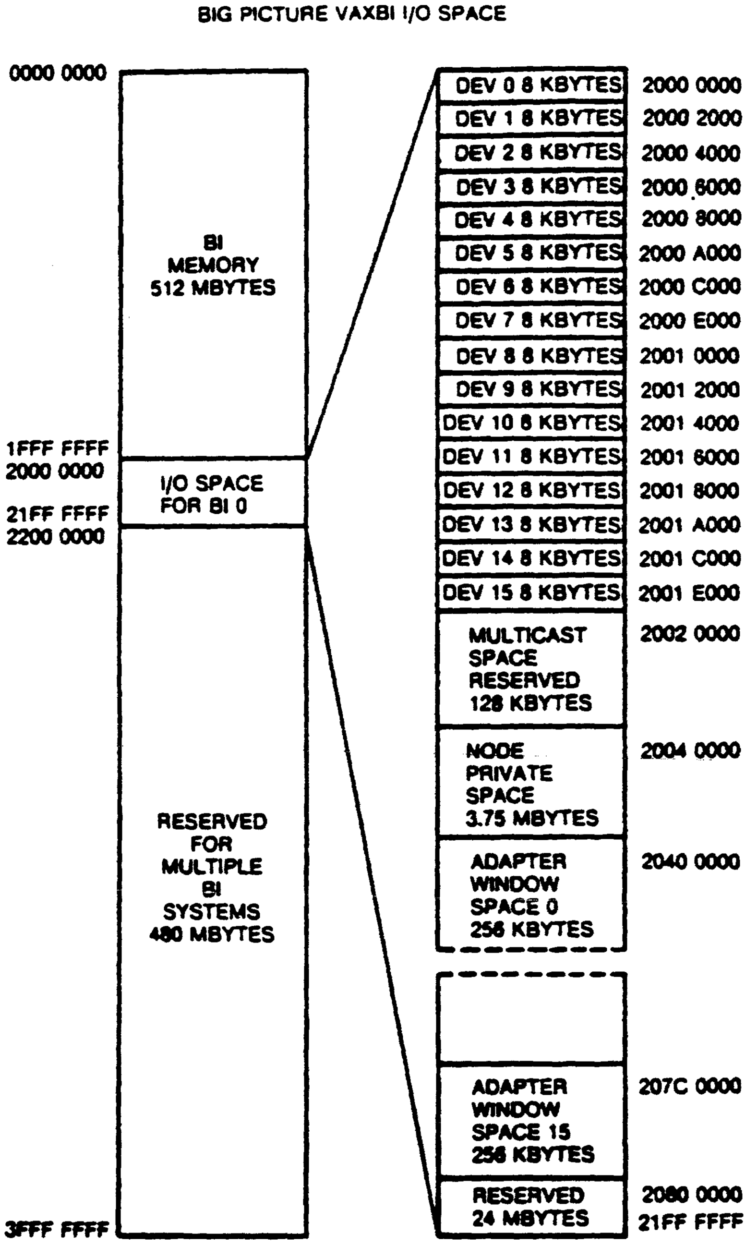


TK-10493

VAX-11/750 Physical Address Space Map

CIBCI Address Assignment

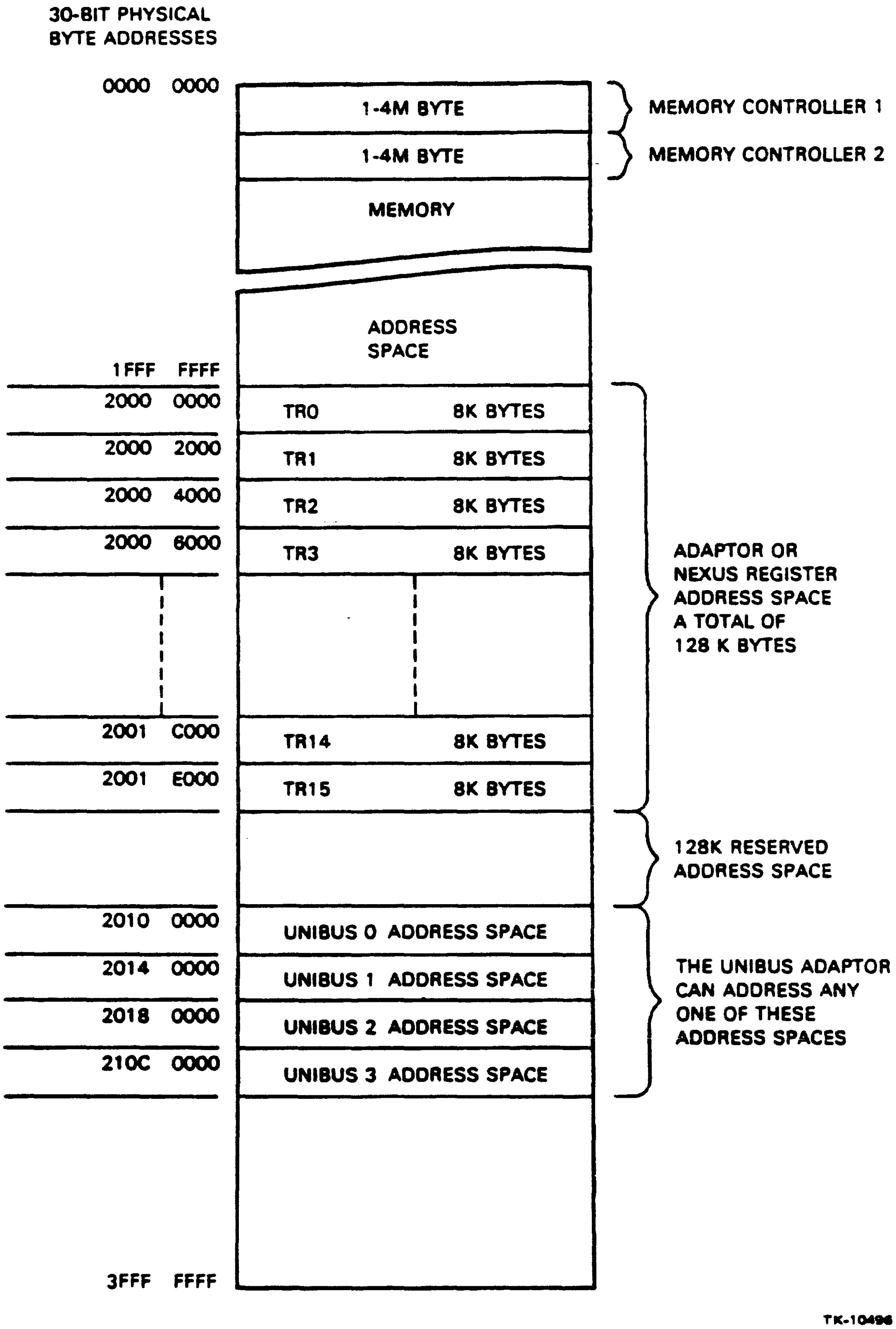
- The address for the CIBCI is a BI node number within the VAXBI I/O space.
- The normal default address is 2000A000, which is BI node 5.



VAXBI I/O Address Space

CI780 Address Assignment

The preferred address for the CI780 within the VAX 780 is at 2001C000 (TR14).



VAX-11/780 Physical Address Space Map

CI780 Address Assignment on the VAX 8600

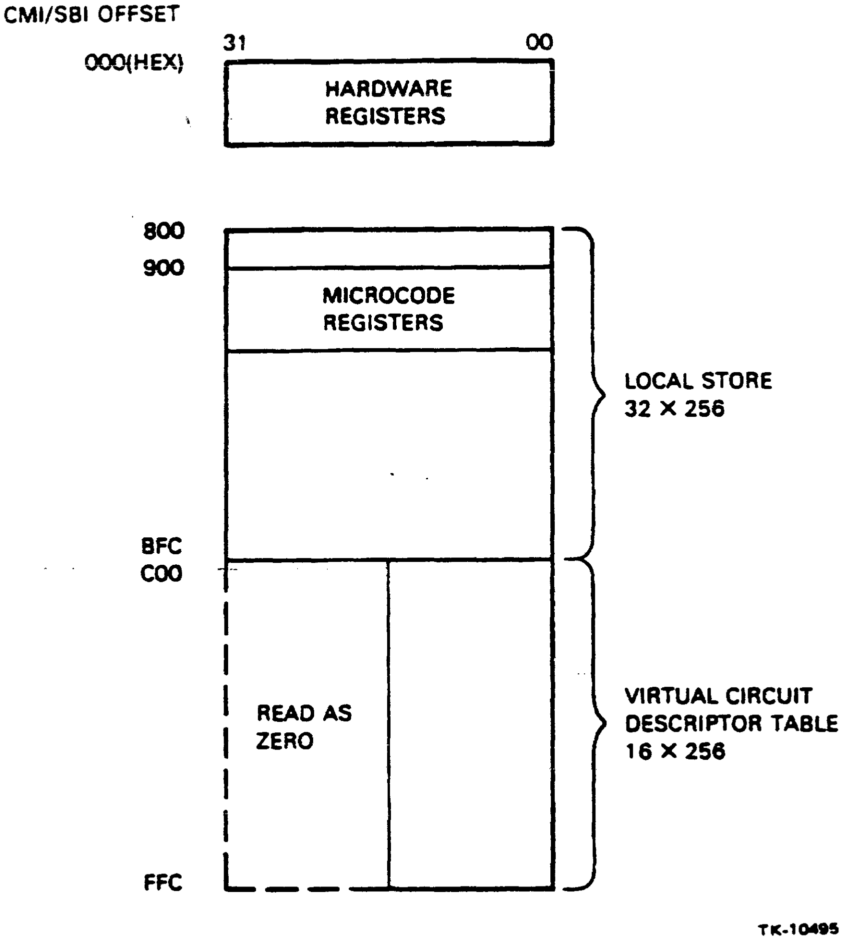
- The CI interface for the VAX 8600 can be attached to one of two SBI interfaces.

- Normal placement is on SBIA 0 with a base address assignment of 2001C000 (same as VAX 780), which is TR #14.

ONLY 1 CI ON 2 SBI MACHINE

		SBIA 0			SBIA 1	
TR#		**Byte Address**	**Longword Address**		**Byte Address**	**Longword Address**
1		20002000	8000800		22002000	8800800
2		20004000	8001000		22004000	8801000
3	DW0	20006000	8001800	DW4	22006000	8801800
4	DW1	20008000	8002000	DW5	22008000	8802000
5	DW2	2000A000	8002800	DW6	2200A000	8802800
6	DW3	2000C000	8003000	DW7	2200C000	8803000
7		2000E000	8003800		2200E000	8803800
8*	RH0	20010000	8004000	RH4	22010000	8804000
9*	RH1	20012000	8004800	RH5	22012000	8804800
10*	RH2	20014000	8005000	RH6	22014000	8805000
11*	RH3	20016000	8005800	RH7	22016000	8805800
12		20018000	8006000		22018000	8806000
13		2001A000	8006800		2201A000	8806800
14		2001C000	8007000		2201C000	8807000
15		2001E000	8007800		2201E000	8807800

*RH780



CI Port Address Space Map

CI Port Hardware Registers

There are four hardware registers common to the CI750, CI780, and CIBCI interfaces:

- Configuration Register (CNFGR) -- located on the SBI Interface module (CI780), CMI Interface module (CI750), and the Adapter Control module (CIBCI).
- Port Maintenance Control and Status Register (PMCSR) -- located on the Data Path module for all three interfaces.
- Maintenance Address Register (MADR) -- located on the Packet Buffer module for all three interfaces.
- Maintenance Data Register (MDATR) -- located on the Packet Buffer module for all three interfaces.

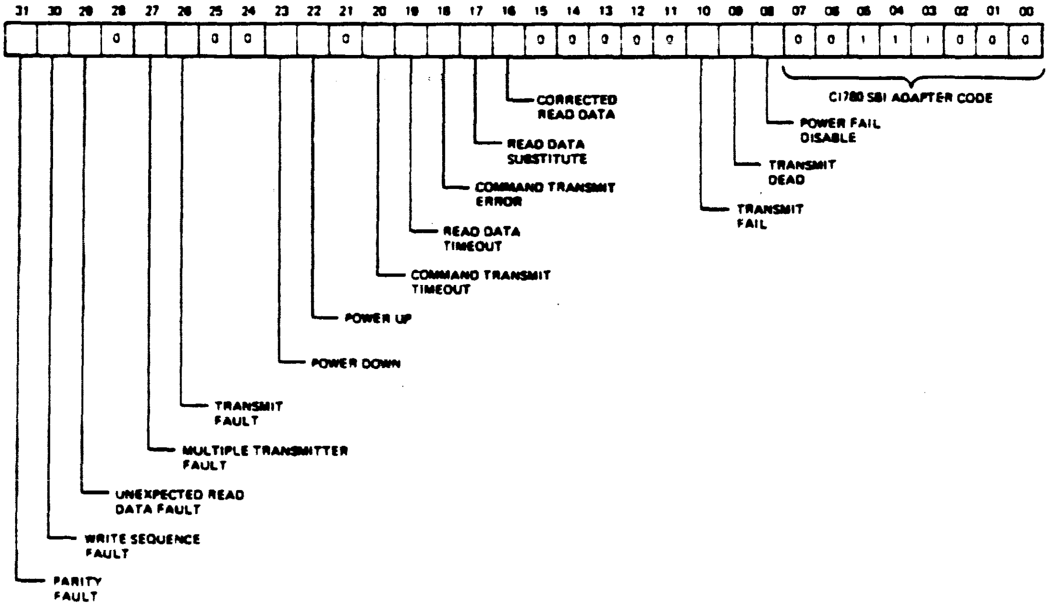
These registers are not part of any memory on the CI Interface. The following diagram shows the byte offsets to each of these registers:

CI750	CI780	CIBCI	HARDWARE REGISTERS
000	000	100	CNFGR
004	004 or 010	110	PMCSR
014	014	114	MADR
018	018	118	MDATR

CIBCA	USES NO CNFGR
N/A	CNFGR
1004	PMCSR
1008	MADR
100C	MDATR

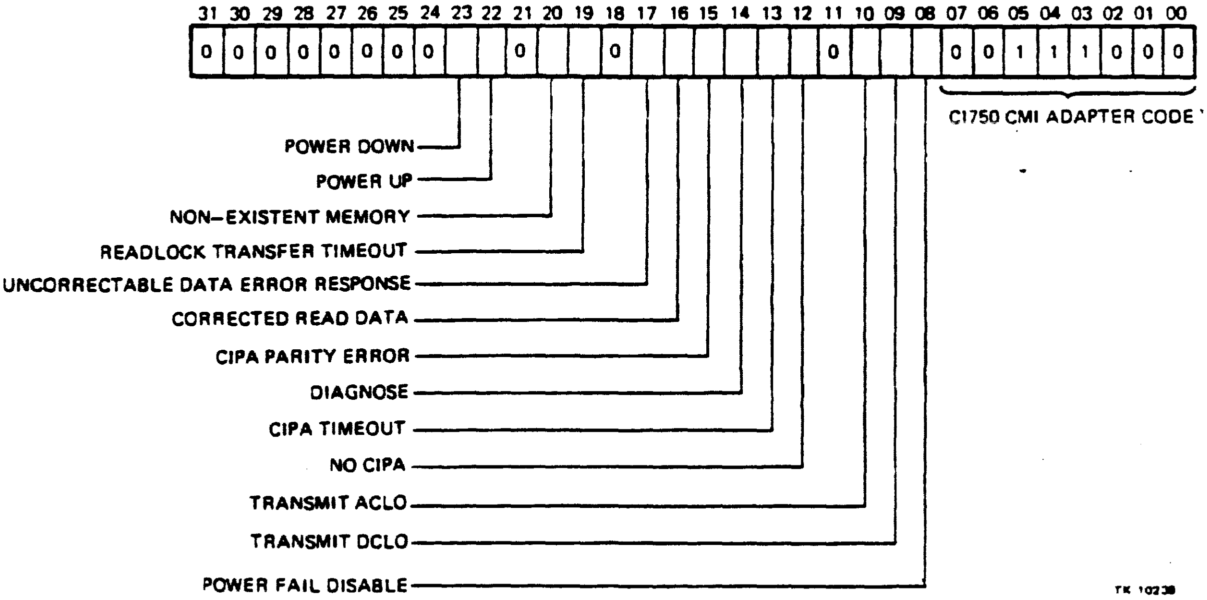
Configuration Register (CNFGR) for CI780

- Byte offset = 0.
- SBI fault bits.
- Port status bits.
- Error bits.



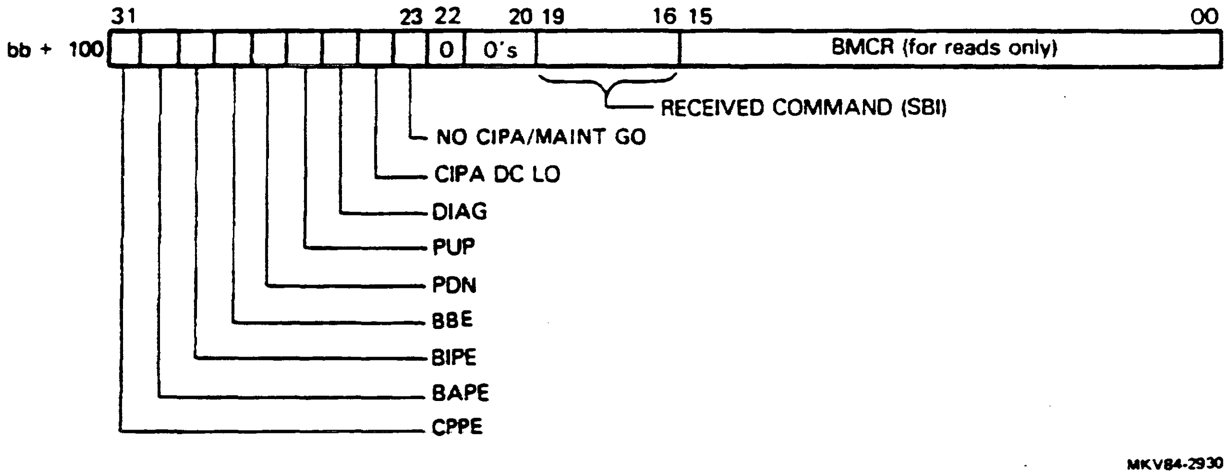
Configuration Register (CNFGR) for CI750

- Byte offset = 0.
- Port status bits.
- Error bits.



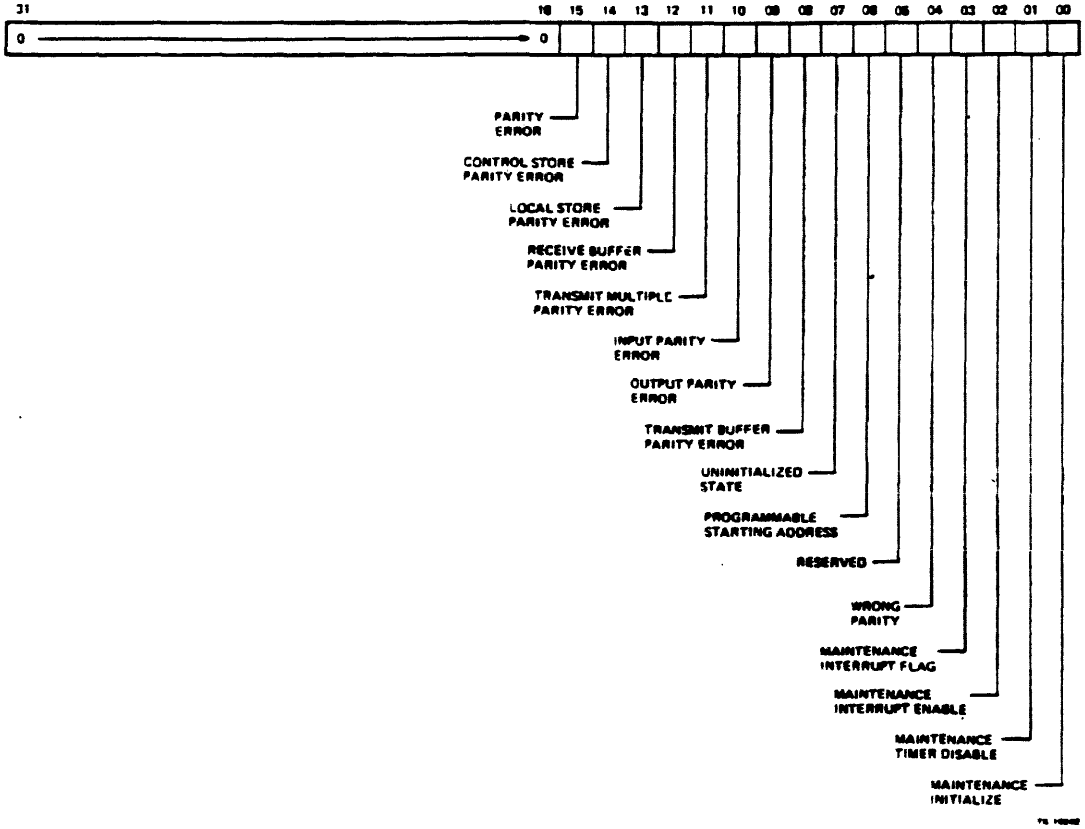
Configuration Register (CNFGR) for CIBCI

- Byte offset = 100.
- Port status bits.
- Error bits.



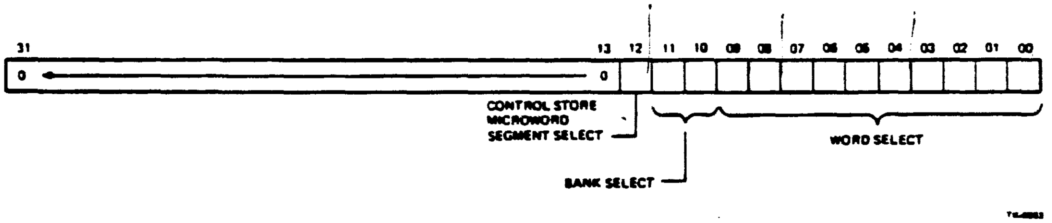
Port Maintenance Control and Status Register (CI750, CI780, CIBCI)

- Port hardware error flags.
- Interrupt and port initialization control bits.
- Byte offset 4 or 10 for CI780.
- Byte offset 4 for CI750.
- Byte offset 110 for CIBCI.
- Bit zero not used in CIBCI.


31
16
15
14
13
12
11
10
09
08
07
06
05
04
03
02
01
00
0
0
PARITY ERROR
CONTROL STORE PARITY ERROR
LOCAL STORE PARITY ERROR
RECEIVE BUFFER PARITY ERROR
TRANSMIT MULTIPLE PARITY ERROR
INPUT PARITY ERROR
OUTPUT PARITY ERROR
TRANSMIT BUFFER PARITY ERROR
UNINITIALIZED STATE
PROGRAMMABLE STARTING ADDRESS
RESERVED
WRONG PARITY
MAINTENANCE INTERRUPT FLAG
MAINTENANCE INTERRUPT ENABLE
MAINTENANCE TIMER DISABLE
MAINTENANCE INITIALIZE

Maintenance Address Register (MADR)

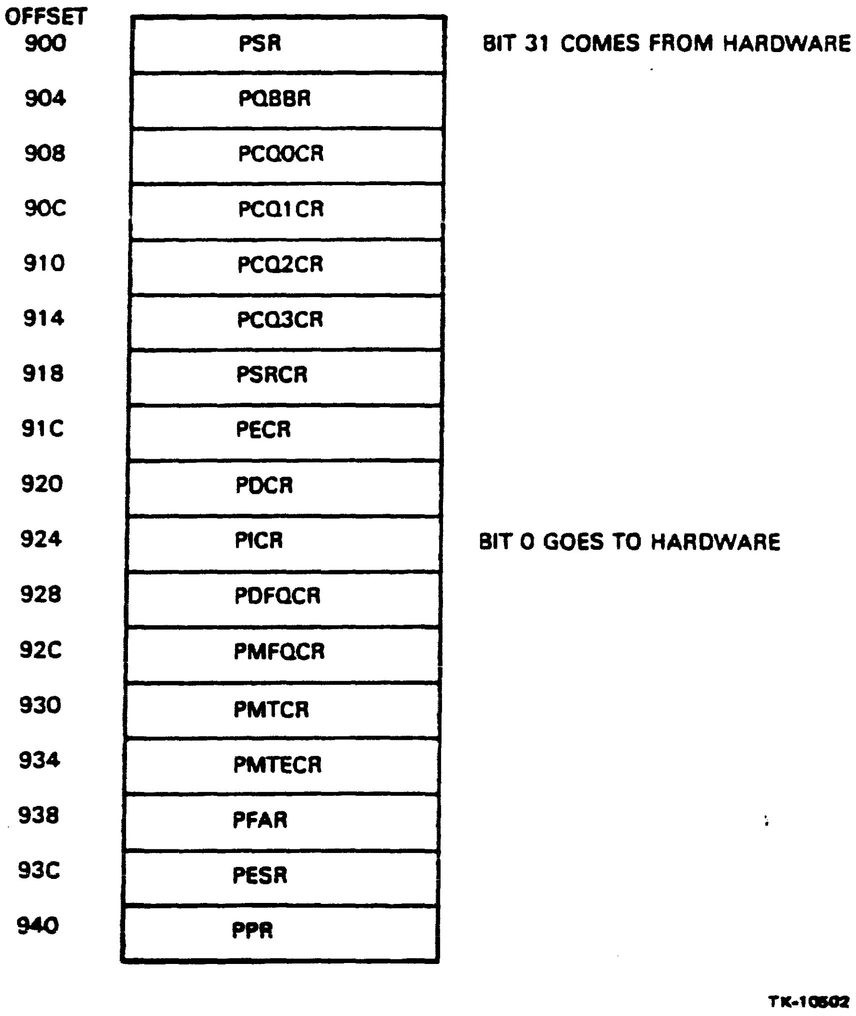
- Byte offset = 14 for CI750 and CI780.
- Byte offset = 114 for CIBCI.
- Addresses the Control Store for loading and verifying the microcode.
- Contains the starting address of the microcode.
- Can only be read/written when port is in uninitialized state (microcode is stopped).



Maintenance Data Register (MDATR)

- Byte offset = 18 for CI750 and CI780.
- Byte offset = 118 for CIBCI.
- Provides access to the Control Store location pointed to by MADR.
- Used to initially load and verify the microcode.
- Valid only when the port in in the uninitialized state (microcode is stopped).

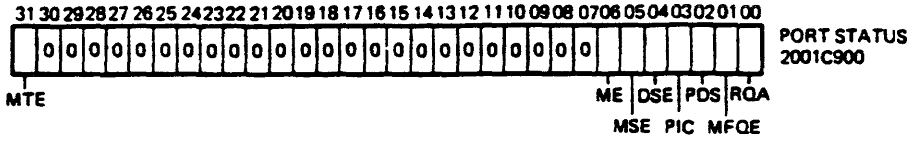
Microcode Registers (in Local Store)



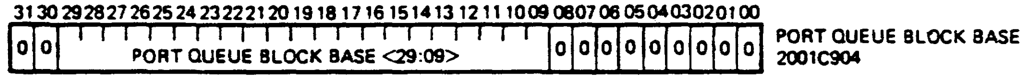
OFFSET		
900	PSR	BIT 31 COMES FROM HARDWARE
904	PQBBR	
908	PCQ0CR	
90C	PCQ1CR	
910	PCQ2CR	
914	PCQ3CR	
918	PSRCR	
91C	PECR	
920	PDCR	
924	PICR	BIT 0 GOES TO HARDWARE
928	PDFQCR	
92C	PMFQCR	
930	PMTCR	
934	PMTECR	
938	PFAR	
93C	PESR	
940	PPR	

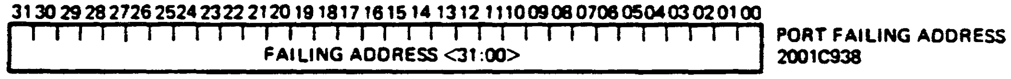
Microcode Register Summary

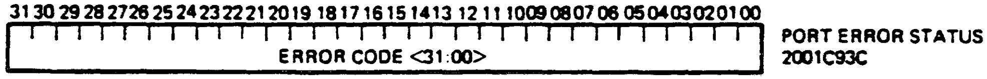
Microcode Registers (in Local Store) (Cont.)

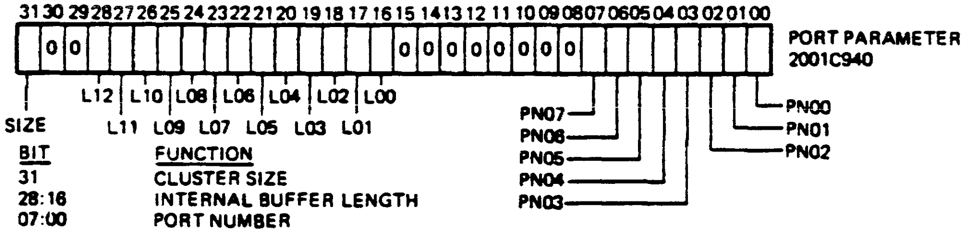


BIT	FUNCTION
31	MAINTENANCE ERROR
06	MAINTENANCE TIMER EXPIRATION
05	MEMORY SYSTEM ERROR
04	DATA STRUCTURE ERROR
03	PORT INITIALIZATION COMPLETE
02	PORT DISABLE COMPLETE
01	MESSAGE FREE QUEUE EMPTY
00	RESPONSE QUEUE AVAILABLE

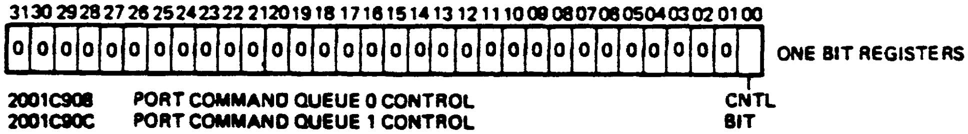








BIT	FUNCTION
31	CLUSTER SIZE
28:16	INTERNAL BUFFER LENGTH
07:00	PORT NUMBER



2001C908	PORT COMMAND QUEUE 0 CONTROL
2001C90C	PORT COMMAND QUEUE 1 CONTROL
2001C910	PORT COMMAND QUEUE 2 CONTROL
2001C914	PORT COMMAND QUEUE 3 CONTROL
2001C918	PORT STATUS RELEASE CONTROL
2001C91C	PORT ENABLE CONTROL
2001C920	PORT DISABLE CONTROL
2001C924	PORT INITIALIZE CONTROL
2001C928	PORT DATAGRAM FREE QUEUE CONTROL
2001C92C	PORT MESSAGE FREE QUEUE CONTROL
2001C930	PORT MAINTENANCE TIMER CONTROL
2001C934	PORT MAINTENANCE TIMER EXPIRATION CONTROL

TK-8538

Microcode Register Breakdown

Device-Specific Registers

CI750 Diagnostic Registers

- Provide diagnostic access to internal circuitry on the CMI interface board.
- Registers

  CMD/ADDR HI
  ADDR LO
  BYTE MASK
  XMIT FILE HI
  XMIT FILE LO
  RECV FILE HI
  RCV FILE LO

CIBCI-Specific Registers

- Registers

  BCI CONTROL
  USER INTERRUPT CONTROL

CI750/CI780/CIBCI INTERNALS

CI Functional Description

Lesson Introduction

This module is a detailed look at the CI Interface. The material is presented in two parts. The first is a general description of each module. The second is a more detailed look at each module.

The CI Interface is functionally divided into five parts: 1) the interface to the system CPU and memory, 2) the Data Paths section, which includes the 2901 ALU, 3) the Transmit and Receive Buffers, 4) the Control Store microsequencer, and 5) the interface to the CI Bus. Flow control through each section is discussed, through the use of block diagrams, with particular attention to the CI Port diagnostics discussed previously.

Lesson Objectives

1. Describe the major functions of the CI Interface and identify which FRU performs each function.
2. Identify the module on which each hardware section resides.
3. Describe the source, destination, function, and error detection method for each of the major buses.
4. Identify the possible loopback paths in the hardware.
5. Describe how CI Bus arbitration is handled.
6. Describe the interaction between the microcode and the hardware.
7. Trace the flow of data through the CI Interface.

Lesson Outline

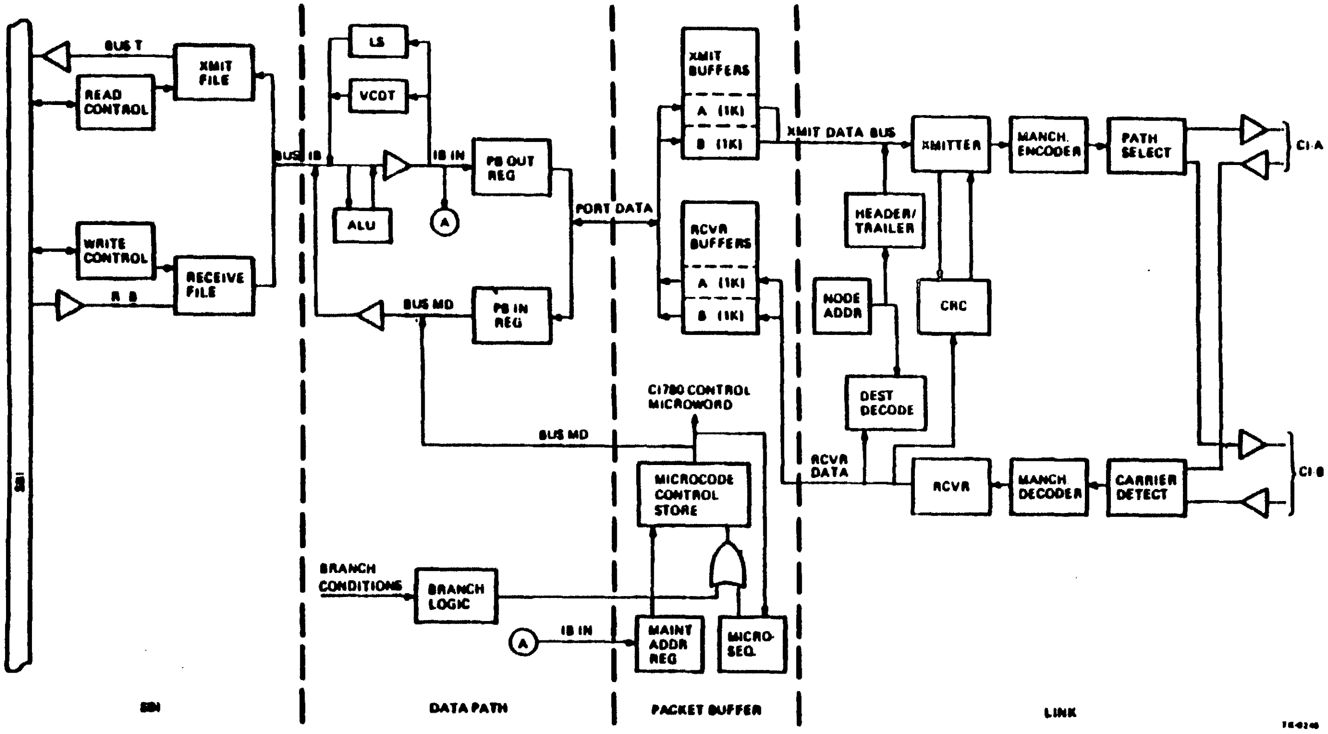
I. General Overview
II. Link Module
III. Packet Buffer Module
IV. Control Store
V. Data Path
VI. System Interface

CI Module Overview

Some of the modules that compose the three different interfaces (CI750, CI780, CIBCI) are identical. The following chart summarizes which modules are shared and which modules are unique to a particular interface:

Module	Part #	CI750	CI780	CIBCI
Link (ILI)	EITHER L0118 L0100	X	X	X
Packet Buffer (PB)	L0101	X	X	X
Data Path (CDP)	L0400	X		X
Data Path (IDP)	L0102		X	
SBI Interface (ISI)	L0104		X	
CMI Interface (CCI)	L0009	X		
BAC	T1017			X
BAD	T1018			X

	CIBCA-A	CIBCA-B
CONTROL	T1015	T1045
LINK	T1025	T1046



CI780 Functional Block Diagram

Link Interface (ILI) L0100 OR L0118

- Parallel-to-serial and serial-to-parallel conversions.
- Manchester encoding (PE).
- Node address switches (true and complement).
- Path selection circuitry.
- Arbitration logic for CI Bus.
- Carrier detection.
- Shared CRC for transmit and receive.
- PAL and ECL logic.
- Data loopback capability for diagnostic purposes.
- Red LED indicates that the internal loopback path is selected but not necessarily in use.
- Green LED indicates external activity on transmit or receive lines.

☆ NOTE — REFERENCE THE HSC INSTALL MANUAL FOR REVISIONS!

(APPENDIX B)

Packet Buffer Module (IPB) L0101

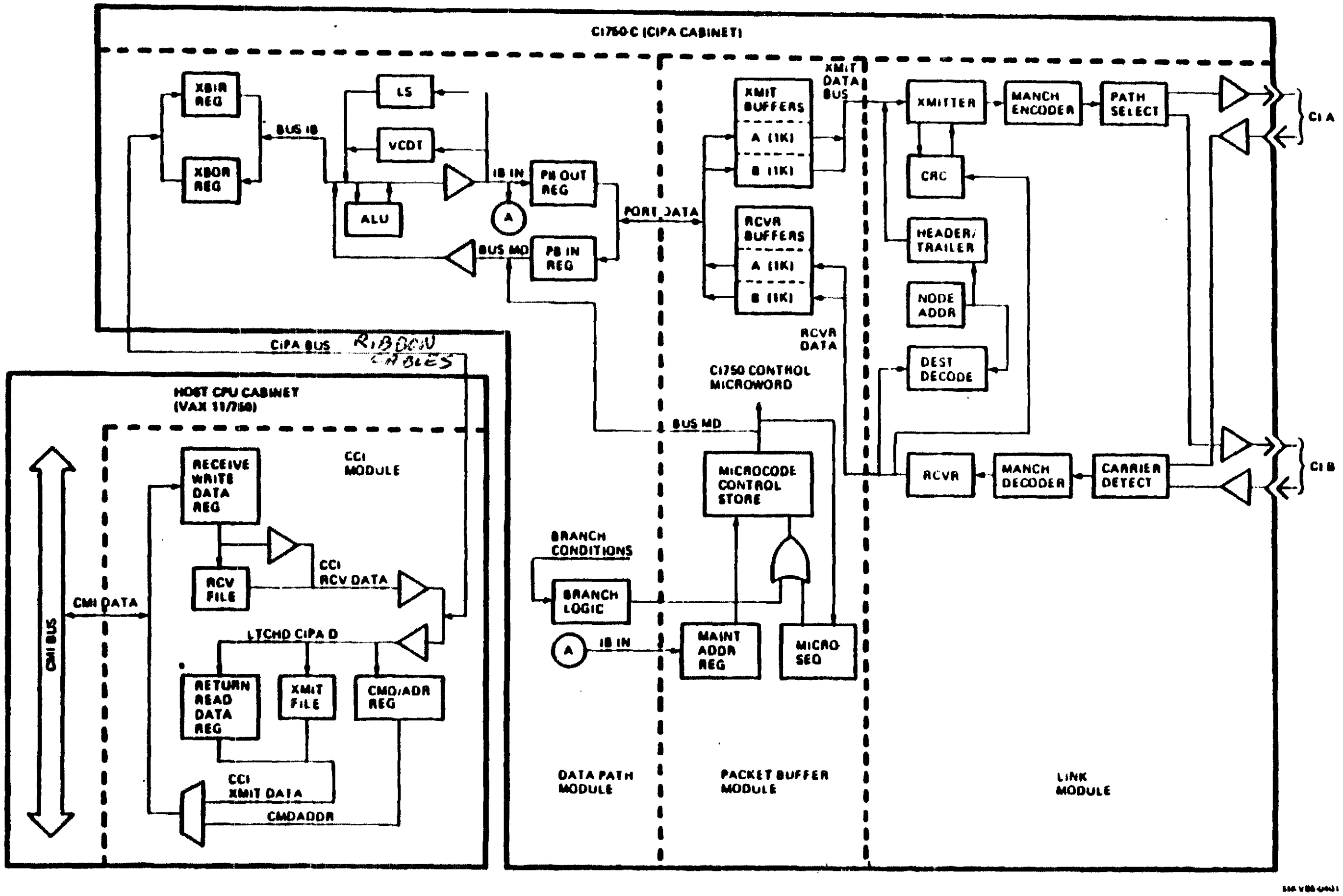
- Microcode Control Store
  - 48 × 3K storage for microcode.
  - 1K is PROM. ☆
  - 2K is RAM.
- Maintenance Access Register
  - Used to specify the address at which to load the microcode.
  - Can be used to specify the starting address for microcode initiation.
- Microsequencer for control of the microcode execution sequence.
- Red LEDs for Control Store address selected.
- Transmit and Receive Buffers
  - Double buffered.
  - Each section holds 1K bytes.
- MADR — MAINT. DATA REG.

Data Path Module (IDP) L0102

- LS (Local Store)
  - 256 × 32 RAM
  - Status storage and scratchpad for microcode.
- VCDT (Virtual Circuit Descriptor Table)
  - 256 × 16 RAM
  - Storage for the status of connections (virtual circuits) between nodes.
- Bus IB provides a 32-bit data path.
- Eight 4-bit 2901 ALUs in parallel make a 32-bit ALU.
- PB OUT and PB IN registers perform 32-to-8 and 8-to-32 bit conversions.
- Microsequencer branch logic located here.

SBI Interface Module (ISI) L0104

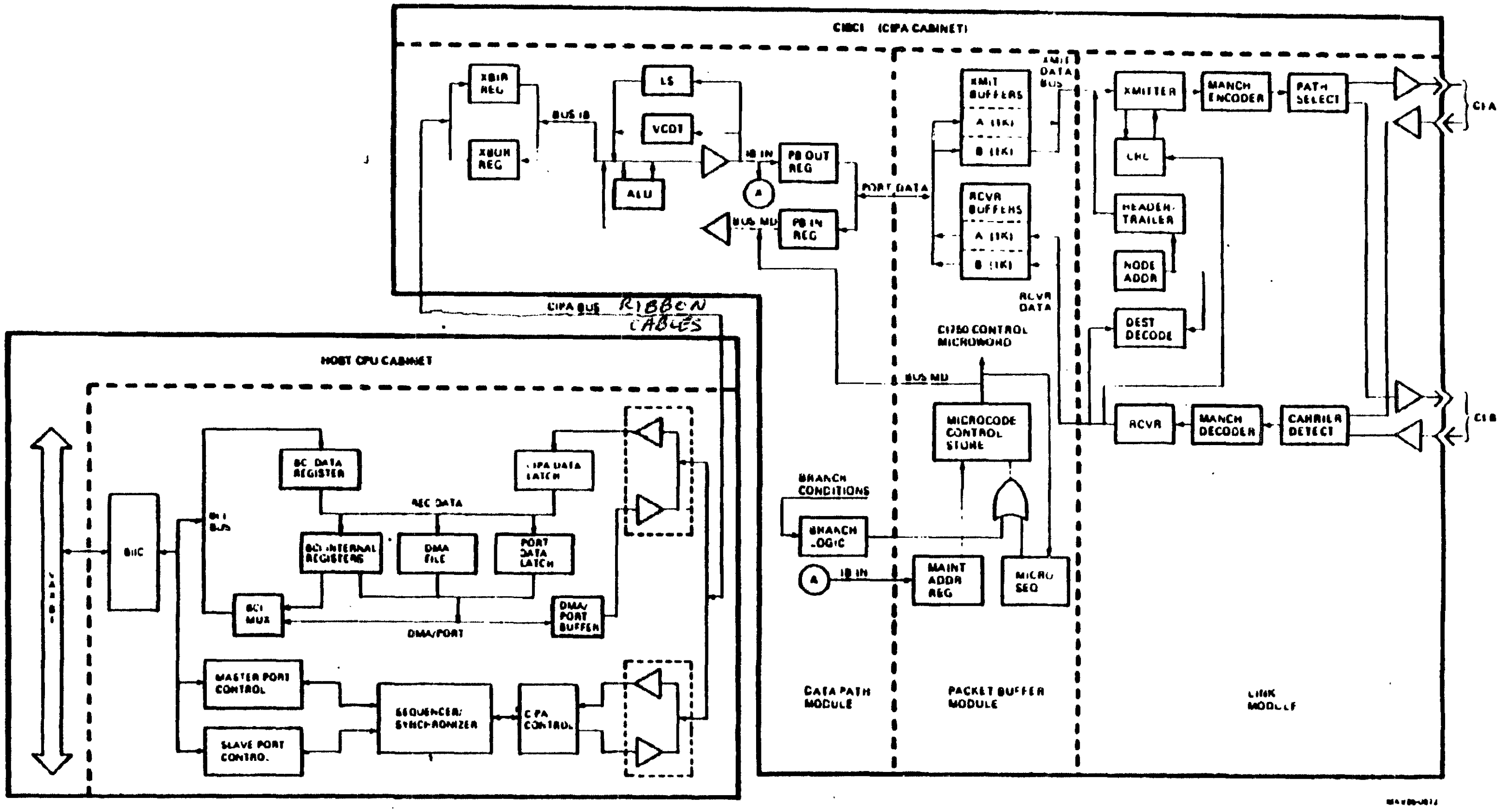
- Interface Section
  - All SBI protocol is implemented here.
  - SBI clock signals provide internal timing for the CI780.
- Transmit and Receive Files
  - Used for packet and data transfers (not for register access).
  - Double buffered
    - a. Two sections (A and B).
    - b. Each section is separate and holds two longwords of data.
    - c. One section can be loaded while the other section is emptied.
- Receive Data Register and Return Read Register
  - Used when accessing registers on the data path module.
  - Each holds one longword at a time.
- Types of Logic
  - Programmed Array Logic (PAL): Smart board
  - Emitter Coupled Logic (ECL)
    - a. Used for SBI clocks (high-speed logic).
    - b. Uses -5 volts supplied from the CPU power supply.
- CONFGR - CONFIG. REGISTER



CI750 Block Diagram

CI750 Module Functions

- Link Interface Module (ILI) L0100 -- same as CI780.
- Packet Buffer Module (IPB) L0101 -- same as CI780.
- Data Path Module (CDP) L0400 -- same as CI780 except XBOR (External Bus Out Register) and XBIR (External Bus In Register) provide a CIPA bus interface to the CMI Interface board.
- CMI Interface Module (CCI) L0009
  - CMI Interface signals.
  - Transmit and receive files.
  - Receive data register and return read register.
  - Controlled by PAL.
  - CIPA bus interface circuitry.

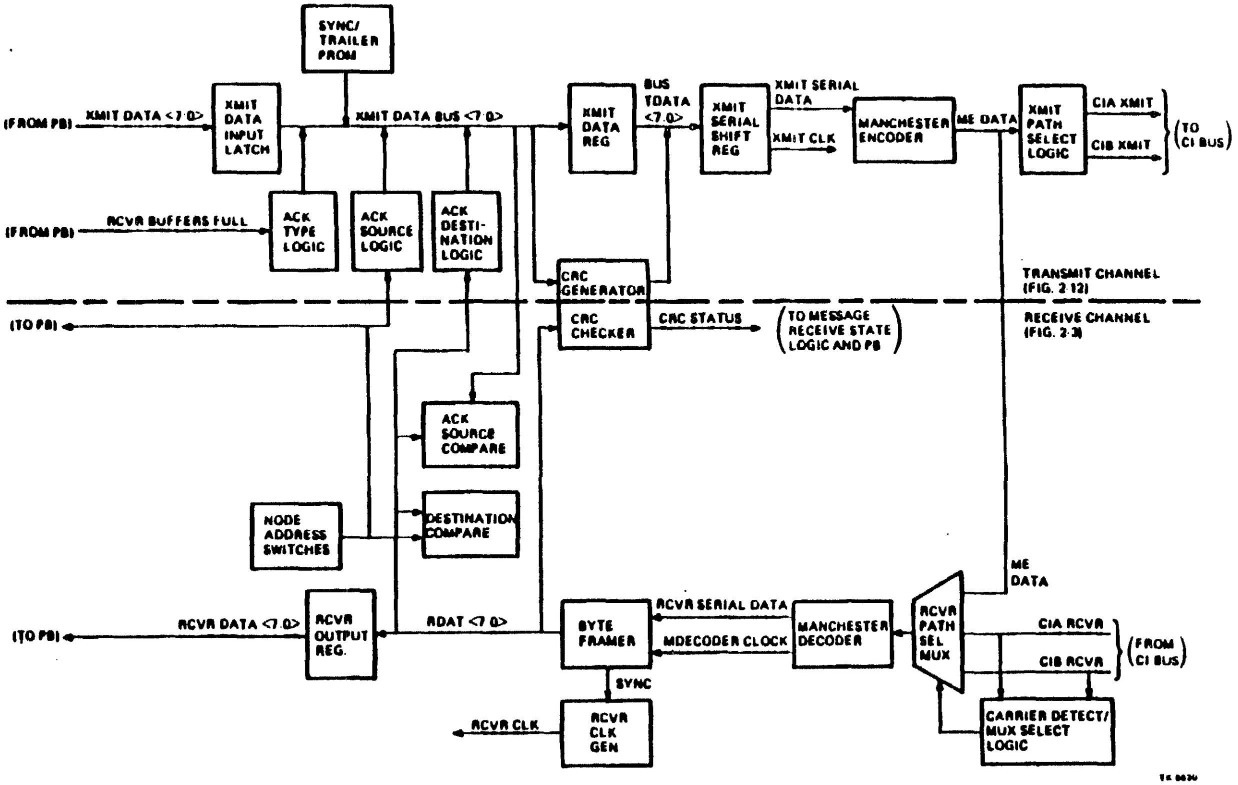


CIBCI Block Diagram

CIBCI Module Functions

- Link Interface Module (ILI) L0100 -- same as CI780.
- Packet Buffer Module (IPB) L0101 -- same as CI780.
- Data Path Module (CDP) L0400 -- same as CI780 except XBOR (External Bus Out Register) and XBIR (External Bus In Register) provide a CIPA bus interface to the BICA Interface boards.
- VAXBI to CIPA Interface (BICA) Modules: BAC and BAD
  - The BIIC (Backplane Interconnect Interface Chip) interfaces the BI bus to the CIPA box through the synchronous BCI (BI Chip Interface) bus.
  - The DMA file and Port Data Latch act as buffers for data passing from BI to CIPA box (or vice versa).
  - The BIIC chip performs all necessary arbitration on the BI bus.
  - The port register contents (on the DP or PB boards) are not read directly, but are loaded into the Port Data Latch for reading, or moved into the Port Data Latch by the BCI side for writing.

THE LINK BOARD



Link Simplified Block Diagram

Link Board Operation

- In order to receive or transmit a packet, the Link board functions can be reduced to four basic operations:

  - Reception of an information packet.
  - Transmission of an ACK/NACK packet.
  - Transmission of an information packet.
  - Reception of an ACK/NACK packet.

- Control of the Link hardware is a function of commands from the Packet Buffer (PB) board (type of operation to perform) and conditions sensed by the logic during the operation.

SLAVE
TO
IPB

- Programmable Array Logic (PAL) chips are used to define/monitor the various hardware states that occur during each of the four basic operations.

- ECL (Emitter-Coupled Logic) voltage levels are used from the XMIT SERIAL SHIFT REG out to the CI cable (on the transmit side) and from the RCVT OUTPUT REG out to the CI cable (on the receive side).

Arbitration for CI Bus

- Based on the countdown of the specific number of "quiet slots" while monitoring for carrier detect:

  TICK = 114nS

  1 quiet slot = 800ns = 7 TICKS

- Number of quiet slots to be counted down determined by the number of nodes attempting to transmit:

  initial count = N + I + 1 where:

  N = 16 (maximum allowable nodes)
  I = node number

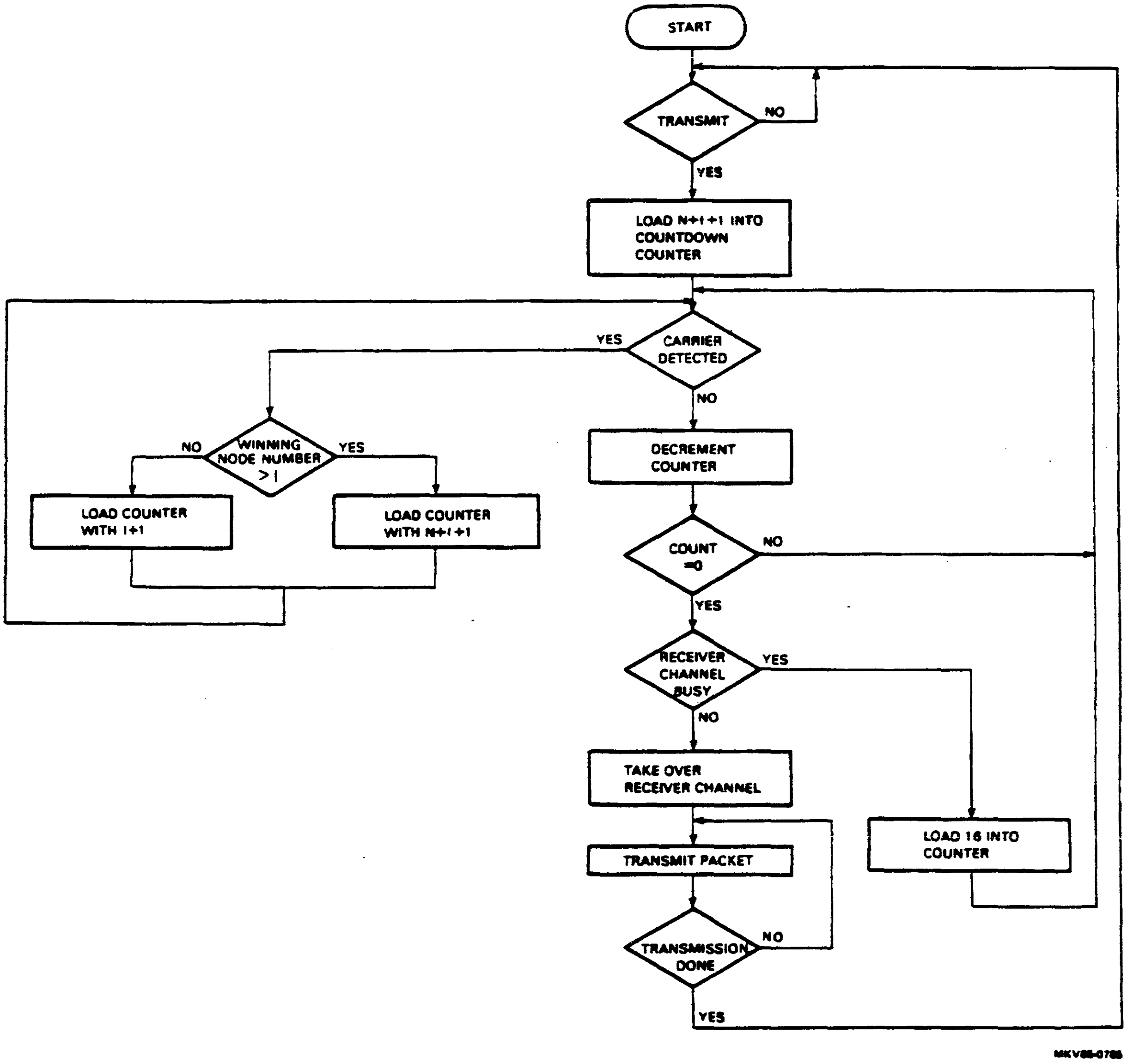
- If a carrier is detected during countdown, the winning node number is determined and following action taken:

  winning node greater, then reset count to (N + I + 1)
  winning node less, then reset count to (I + 1)

- Allows competing nodes access to the bus with the lowest-numbered node having the highest priority.

  ↘ THIS IS WHY HSC's @ LOWEST #

☆ REFERENCE
PAGE 2-38
IN CIBCI TECH. DESCRIPT MAN

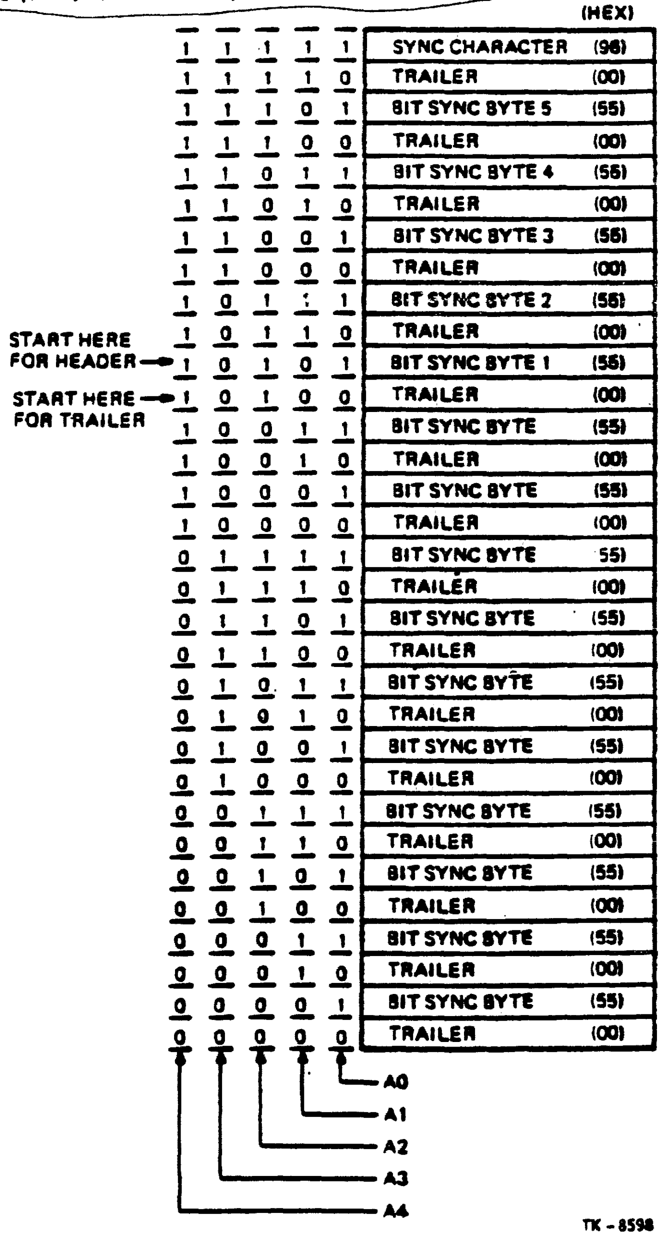


Arbitration Flow Diagram

Sync/Trailer PROM Space

- Used during transmit operation, controlled by transmit control logic.
- Address line A0 determines type of PROM output (sync or trailer) while lines A4:A2 change (count) to determine the correct number of sync/trailer bytes.
- Lower area of PROM reserved for possible expansion.

NEVER USED

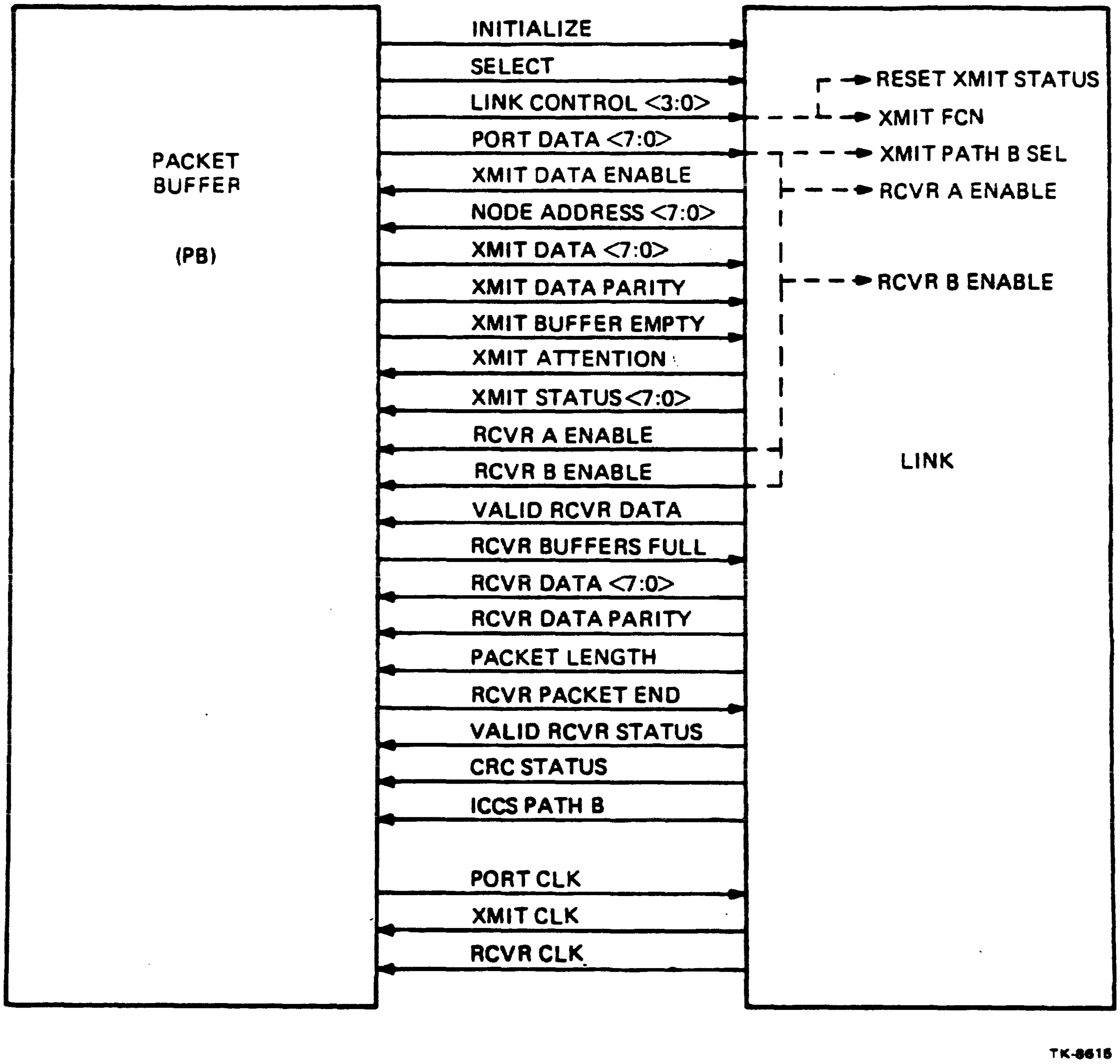


A4	A3	A2	A1	A0		(HEX)
1	1	1	1	1	SYNC CHARACTER	(96)
1	1	1	1	0	TRAILER	(00)
1	1	1	0	1	BIT SYNC BYTE 5	(55)
1	1	1	0	0	TRAILER	(00)
1	1	0	1	1	BIT SYNC BYTE 4	(56)
1	1	0	1	0	TRAILER	(00)
1	1	0	0	1	BIT SYNC BYTE 3	(56)
1	1	0	0	0	TRAILER	(00)
1	0	1	1	1	BIT SYNC BYTE 2	(56)
1	0	1	1	0	TRAILER	(00)
1	0	1	0	1	BIT SYNC BYTE 1	(56)
1	0	1	0	0	TRAILER	(00)
1	0	0	1	1	BIT SYNC BYTE	(55)
1	0	0	1	0	TRAILER	(00)
1	0	0	0	1	BIT SYNC BYTE	(55)
1	0	0	0	0	TRAILER	(00)
0	1	1	1	1	BIT SYNC BYTE	55)
0	1	1	1	0	TRAILER	(00)
0	1	1	0	1	BIT SYNC BYTE	(55)
0	1	1	0	0	TRAILER	(00)
0	1	0	1	1	BIT SYNC BYTE	(55)
0	1	0	1	0	TRAILER	(00)
0	1	0	0	1	BIT SYNC BYTE	(55)
0	1	0	0	0	TRAILER	(00)
0	0	1	1	1	BIT SYNC BYTE	(55)
0	0	1	1	0	TRAILER	(00)
0	0	1	0	1	BIT SYNC BYTE	(55)
0	0	1	0	0	TRAILER	(00)
0	0	0	1	1	BIT SYNC BYTE	(55)
0	0	0	1	0	TRAILER	(00)
0	0	0	0	1	BIT SYNC BYTE	(55)
0	0	0	0	0	TRAILER	(00)

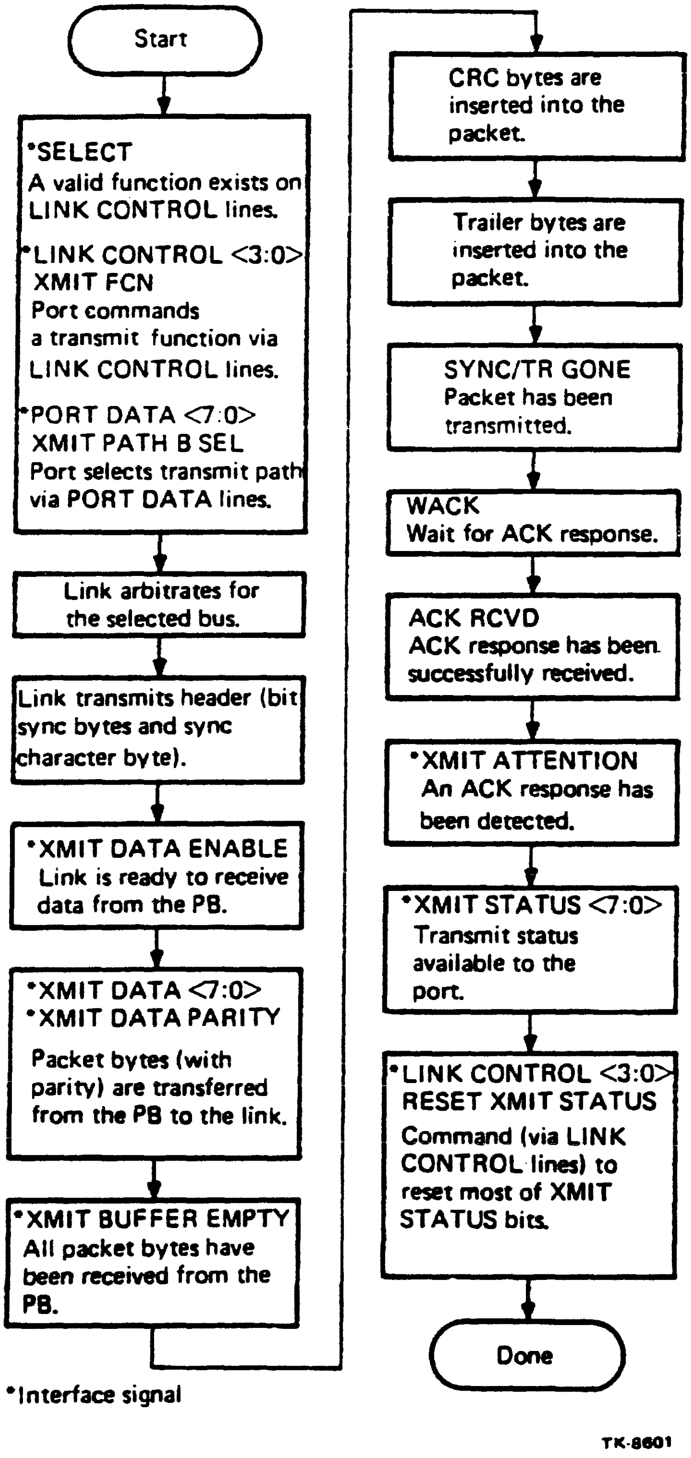
Sync/Trailer PROM Space

Link Interface Signals

The following three figures illustrate how the Link board is controlled by signals from the Packet Buffer board. The first diagram shows the Link/Packet Buffer interface signals. The flow diagrams highlight these signals to illustrate their role in a typical transmit or receive operation.

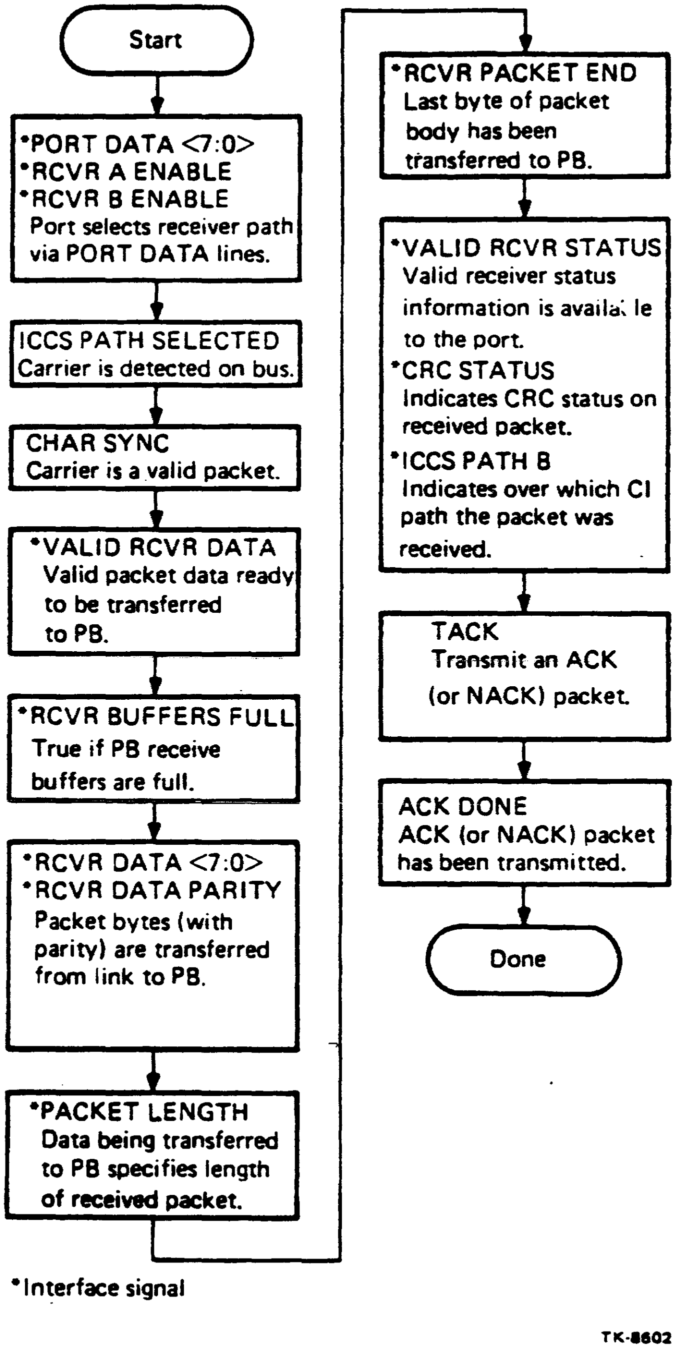


Link Interface Signals


Start
*SELECT
A valid function exists on LINK CONTROL lines.
*LINK CONTROL <3:0>
XMIT FCN
Port commands a transmit function via LINK CONTROL lines.
*PORT DATA <7:0>
XMIT PATH B SEL
Port selects transmit path via PORT DATA lines.
Link arbitrates for the selected bus.
Link transmits header (bit sync bytes and sync character byte).
*XMIT DATA ENABLE
Link is ready to receive data from the PB.
*XMIT DATA <7:0>
*XMIT DATA PARITY
Packet bytes (with parity) are transferred from the PB to the link.
*XMIT BUFFER EMPTY
All packet bytes have been received from the PB.
CRC bytes are inserted into the packet.
Trailer bytes are inserted into the packet.
SYNC/TR GONE
Packet has been transmitted.
WACK
Wait for ACK response.
ACK RCVD
ACK response has been successfully received.
*XMIT ATTENTION
An ACK response has been detected.
*XMIT STATUS <7:0>
Transmit status available to the port.
*LINK CONTROL <3:0>
RESET XMIT STATUS
Command (via LINK CONTROL lines) to reset most of XMIT STATUS bits.
Done
*Interface signal
TK-8601

Interface Flow Diagram -- Transmit Operation

Link Interface Signals (Cont.)


Start
*PORT DATA <7:0>
*RCVR A ENABLE
*RCVR B ENABLE
Port selects receiver path via PORT DATA lines.
ICCS PATH SELECTED
Carrier is detected on bus.
CHAR SYNC
Carrier is a valid packet.
*VALID RCVR DATA
Valid packet data ready to be transferred to PB.
*RCVR BUFFERS FULL
True if PB receive buffers are full.
*RCVR DATA <7:0>
*RCVR DATA PARITY
Packet bytes (with parity) are transferred from link to PB.
*PACKET LENGTH
Data being transferred to PB specifies length of received packet.
*RCVR PACKET END
Last byte of packet body has been transferred to PB.
*VALID RCVR STATUS
Valid receiver status information is availa. le to the port.
*CRC STATUS
Indicates CRC status on received packet.
*ICCS PATH B
Indicates over which CI path the packet was received.
TACK
Transmit an ACK (or NACK) packet.
ACK DONE
ACK (or NACK) packet has been transmitted.
Done
*Interface signal
TK-8602

Interface Flow Diagram -- Receive Operation

THE PACKET BUFFER BOARD

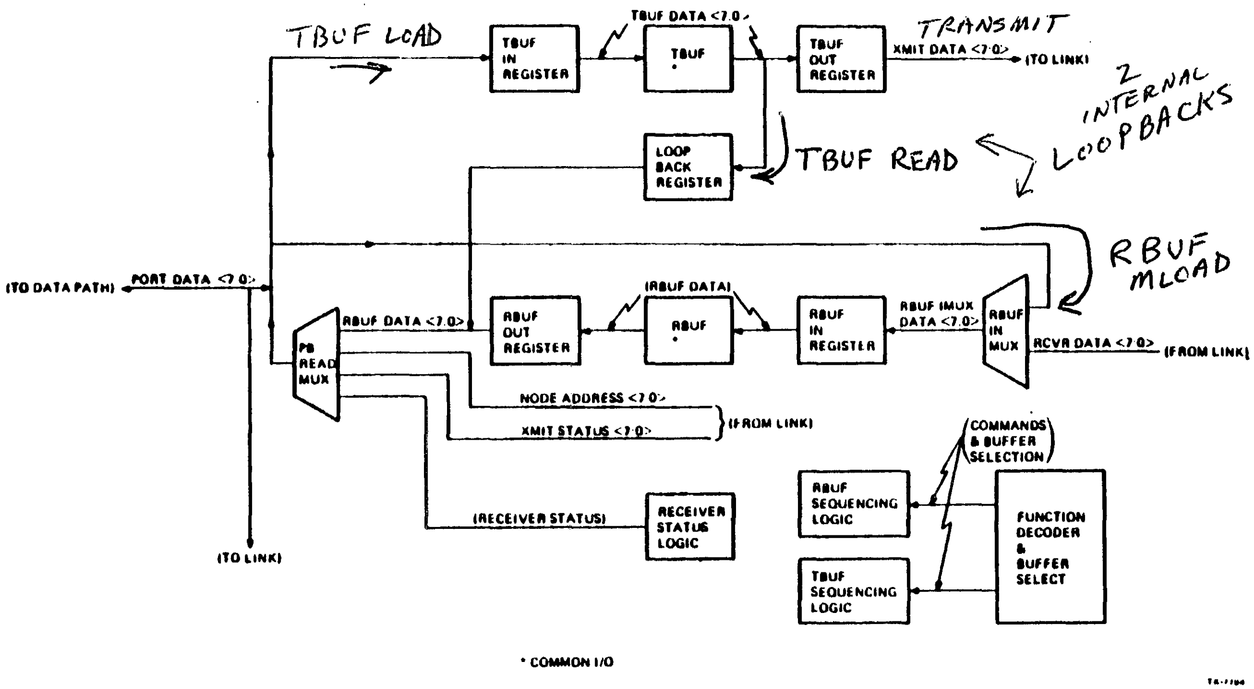
Packet Buffer Board Operation

- All CI-bound packets flow through the PB board
- Transmit buffer (TBUF) contains an A and a B section, each section being 1Kb in size.
- Receive buffer (RBUFF) contains an A and a B section, each section being 1Kb in size.
- Buffers are loaded and read under control of port microcode, all transfers can be reduced to six operations:

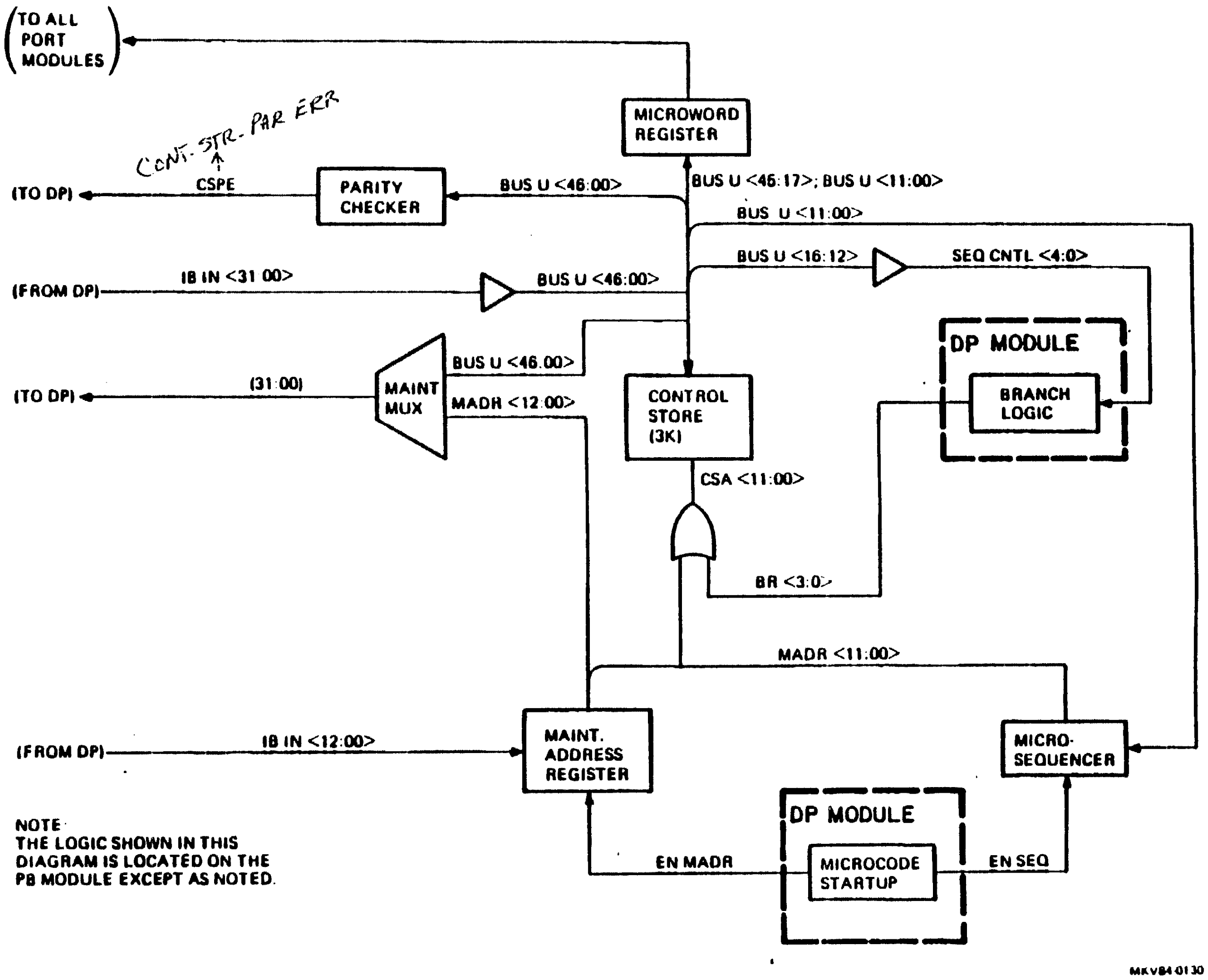
  TBUF load
  Transmit
  RBUF load
  RBUF read
  TBUF read
  RBUF MLOAD

- Addressing on RBUF is controlled by the counter derived from the "packet length" section of the received packet.

JUST BUFFER SECTION SHOWN

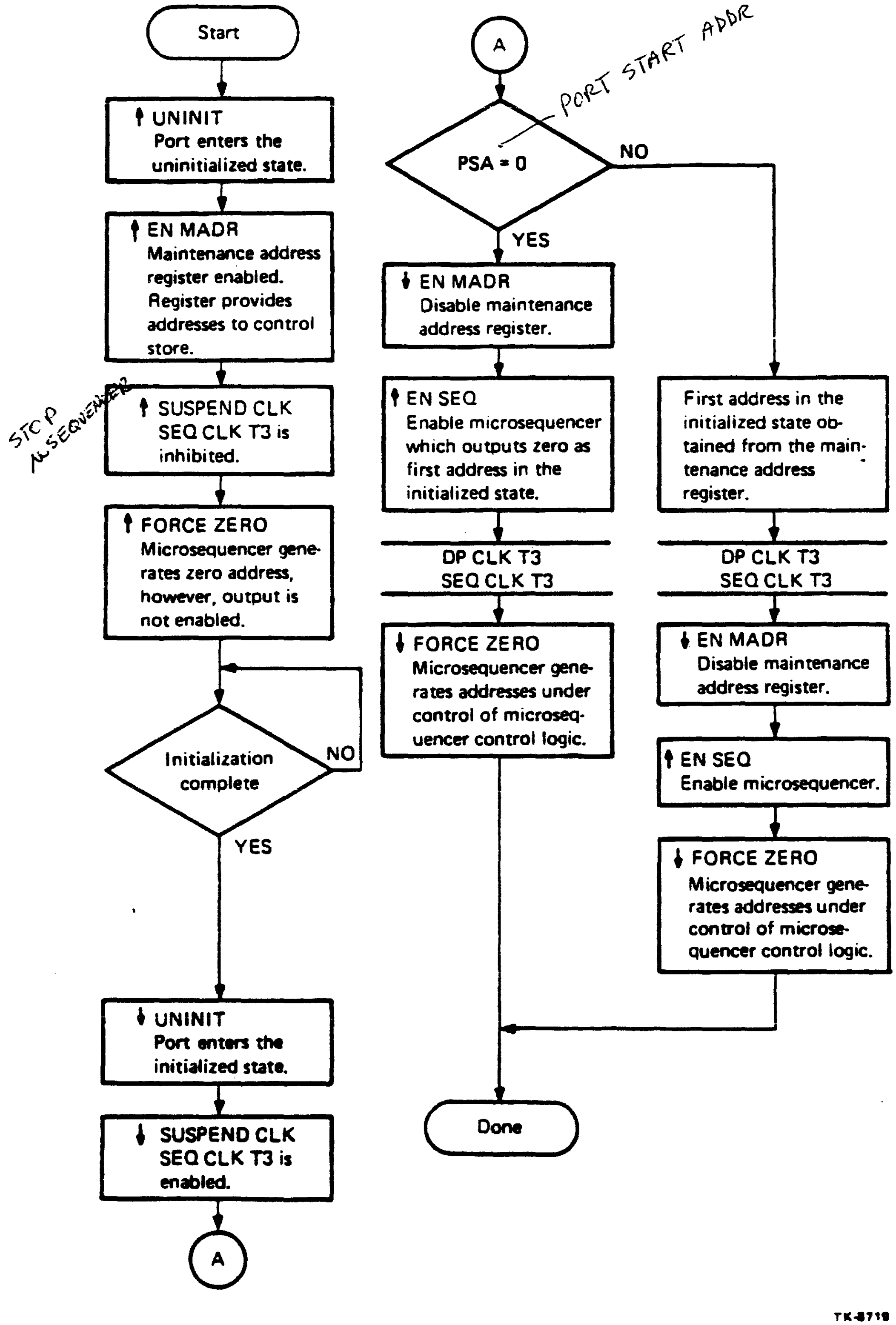


Packet Buffer Data Flow

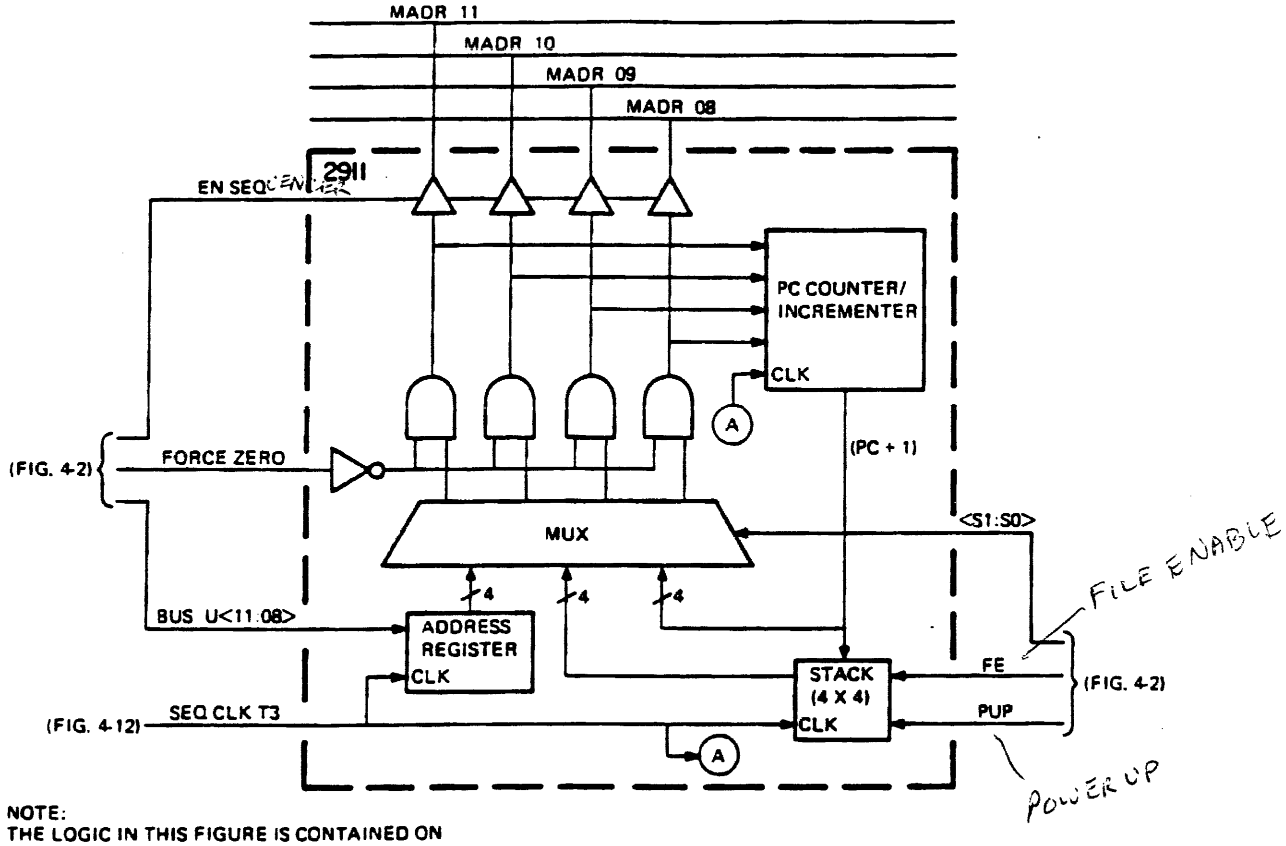


NOTE:
THE LOGIC SHOWN IN THIS DIAGRAM IS LOCATED ON THE PB MODULE EXCEPT AS NOTED.

Control Store Simplified Block Diagram


Start
↑ UNINIT
Port enters the uninitialized state.
↑ EN MADR
Maintenance address register enabled. Register provides addresses to control store.
STOP µSEQUENCER
↑ SUSPEND CLK
SEQ CLK T3 is inhibited.
↑ FORCE ZERO
Microsequencer generates zero address, however, output is not enabled.
Initialization complete
NO
YES
↓ UNINIT
Port enters the initialized state.
↓ SUSPEND CLK
SEQ CLK T3 is enabled.
A
A
PORT START ADDR
PSA = 0
NO
YES
↓ EN MADR
Disable maintenance address register.
↑ EN SEQ
Enable microsequencer which outputs zero as first address in the initialized state.
DP CLK T3
SEQ CLK T3
↓ FORCE ZERO
Microsequencer generates addresses under control of microsequencer control logic.
First address in the initialized state obtained from the maintenance address register.
DP CLK T3
SEQ CLK T3
↓ EN MADR
Disable maintenance address register.
↑ EN SEQ
Enable microsequencer.
↓ FORCE ZERO
Microsequencer generates addresses under control of microsequencer control logic.
Done
TK-8719

Microcode Start-Up Flow Diagram

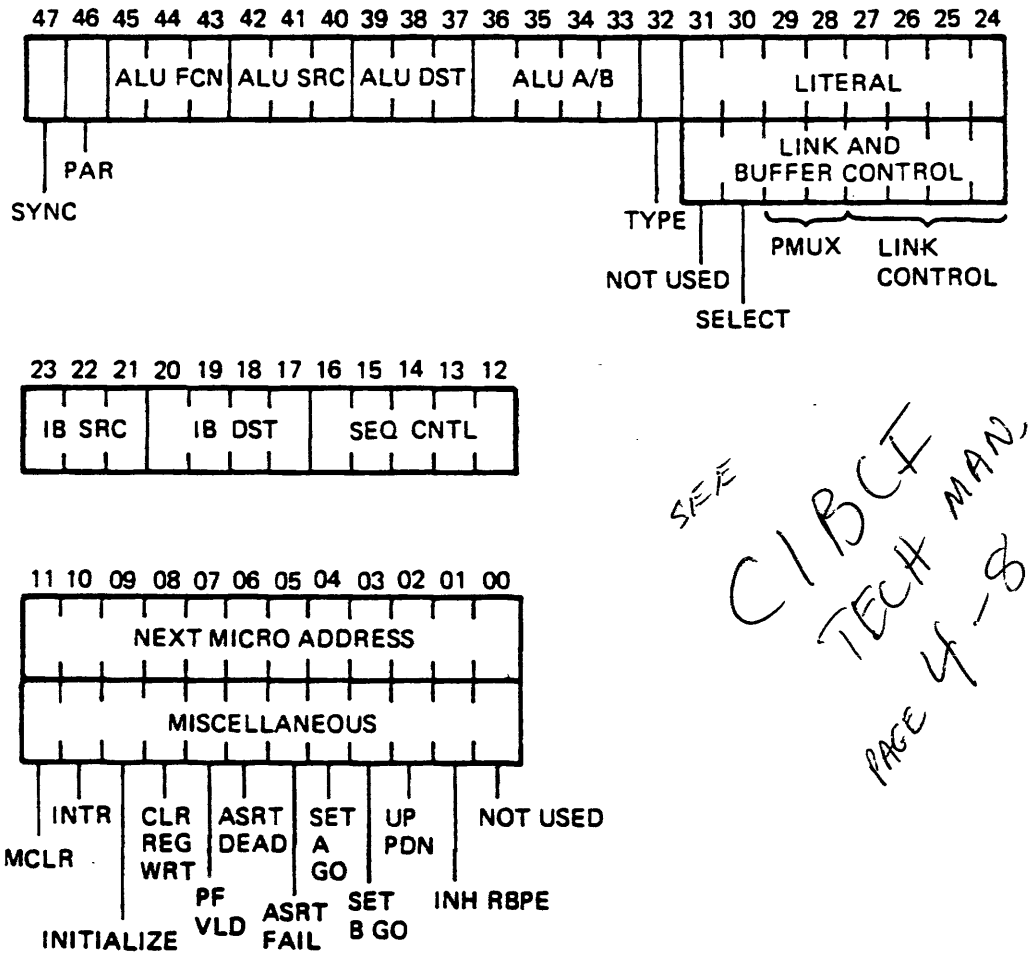


NOTE:
THE LOGIC IN THIS FIGURE IS CONTAINED ON SHEET P OF THE PB ENGINEERING DRAWINGS.

The 2911 Microsequencer

The Microword

Output from the Control Store is a 48-bit microword, most of which passes into the microword register. The output of the microword register provides control signals to all of the port modules as well as address values for the next microinstruction.



SEE CIBCI TECH MAN, PAGE 4-8

TK-8720

Microword Fields

THE DATA PATH BOARD

Data Path Board Operation

- Contains three 32-bit wide buses:

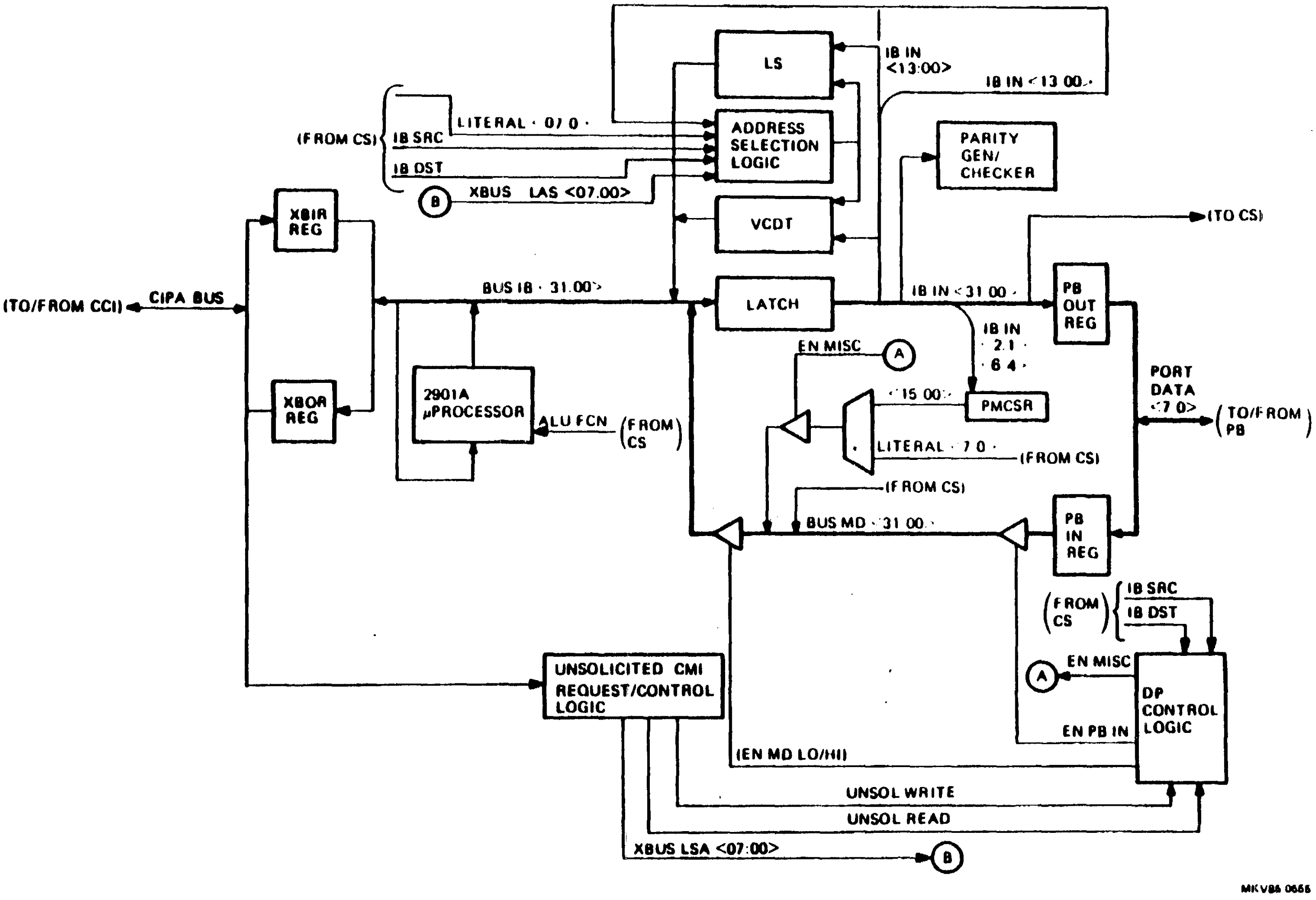
  IB Bus (internal bus)
  MD Bus (miscellaneous data)
  IB IN Bus

- Data sources for the IB:

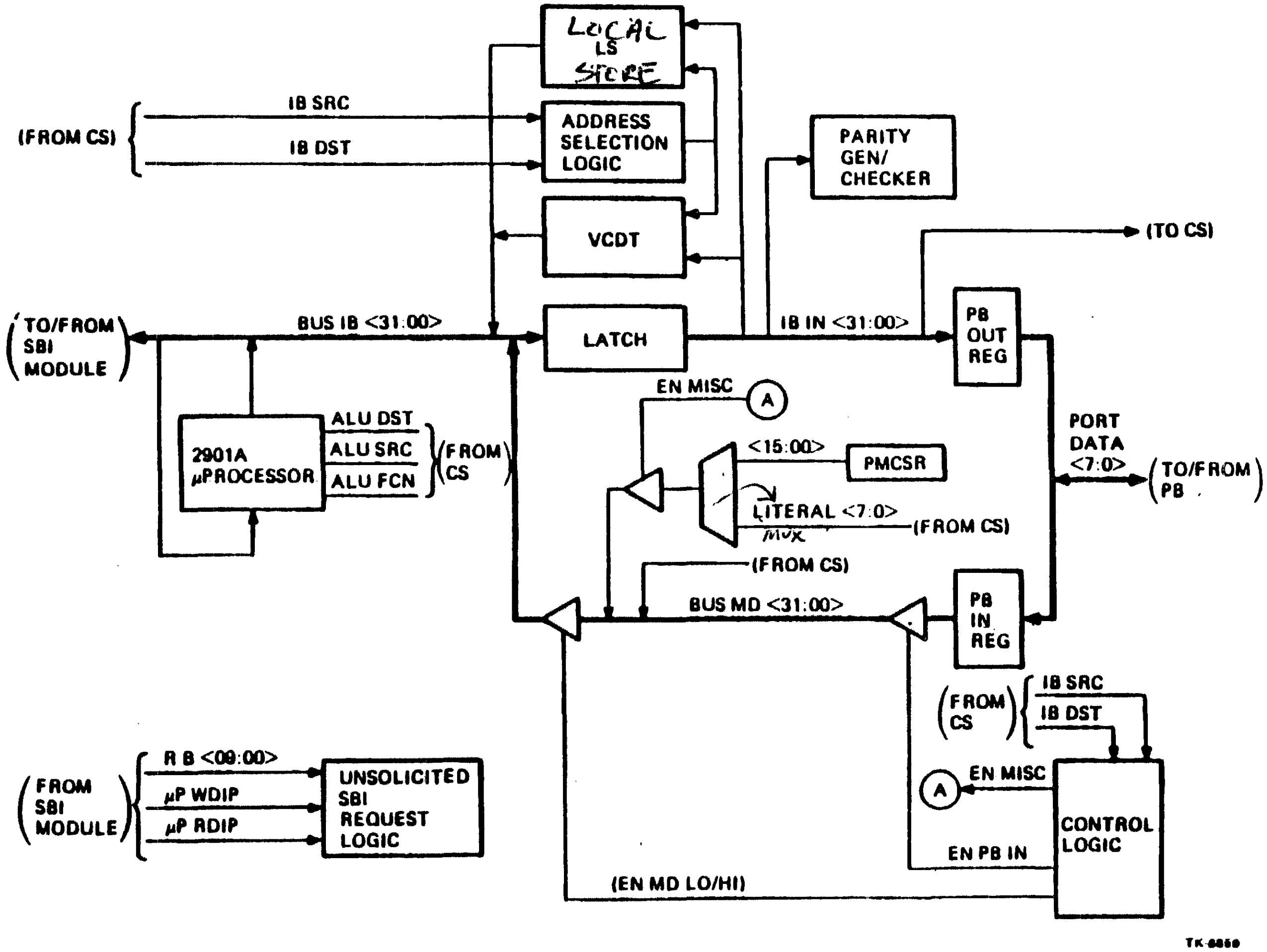
  LS RAM (Local Store)
  VCDT RAM Circuit Descriptor Table)
  2901A ALU
  Microword from CS (Control Store)
  MADR
  PMCSR
  PB IN REG
  Microword literal field
  SBI Interface or XBIR Register (depending on the type of CI Interface being used)

- Local Store is 256 × 32 RAM VCDT is 256 × 16 RAM

- When the port is under microword control, the IB DST field and the IB SRC field select internal bus (IB) destination and source respectively.



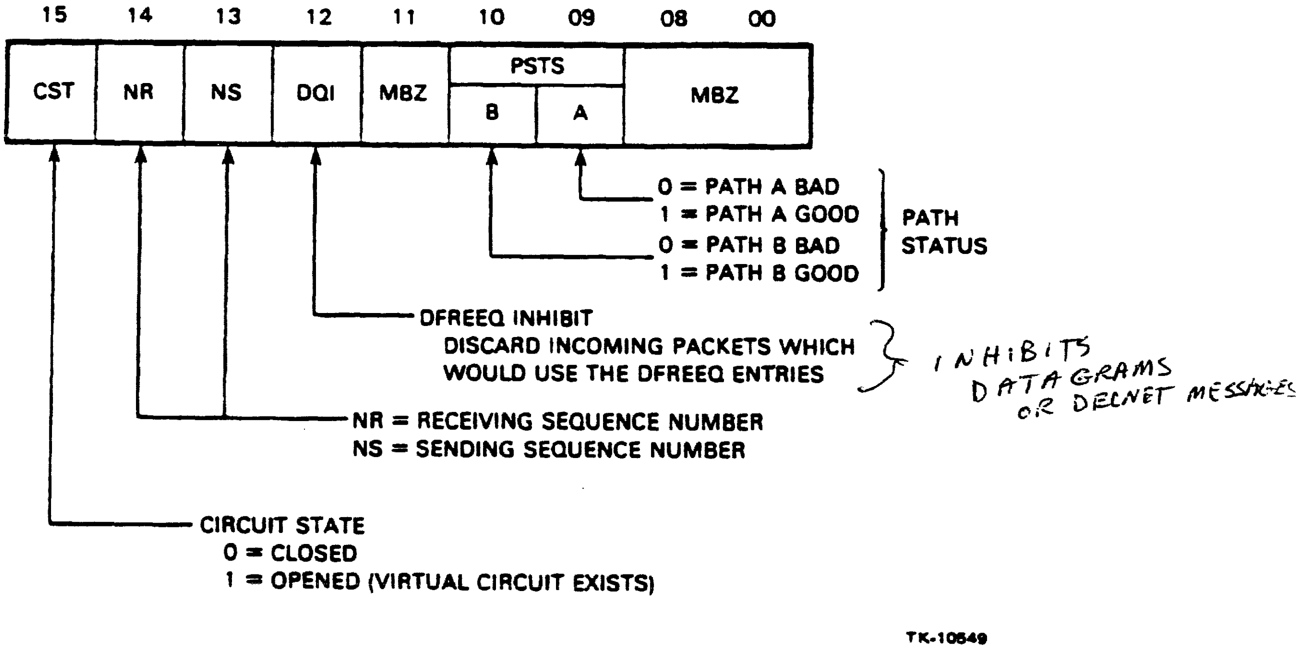
Data Path Module Block Diagram (CI750, CIBCI)



Data Path Block Diagram (CI780)

The Virtual Circuit Descriptor Table

The VCDT contains important information used by the high-level cluster software (connection manager) to direct/coordinate packet transfer throughout the cluster. It is actually built by the port driver and microcode working together. Here is a sample entry (one per node):

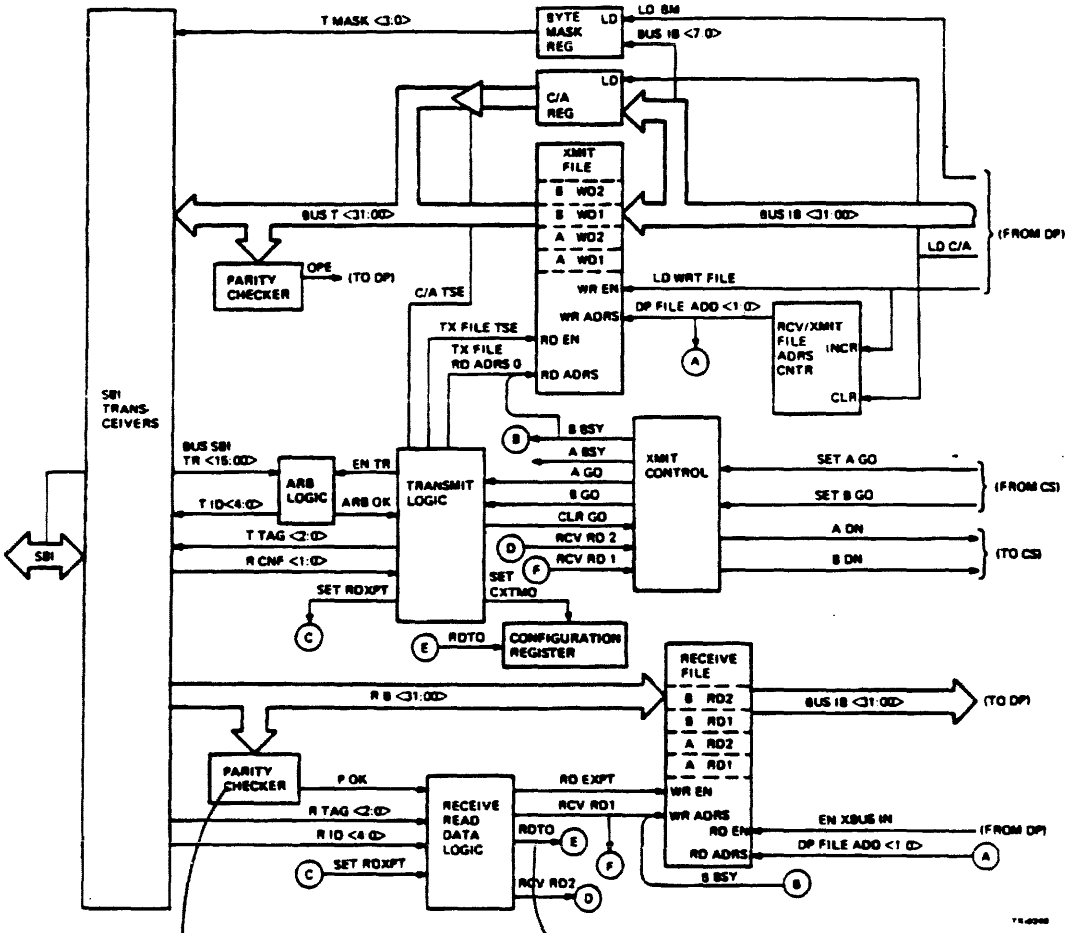

15
14
13
12
11
10
09
08
00
CST
NR
NS
DQI
MBZ
PSTS
B
A
MBZ
0 = PATH A BAD
1 = PATH A GOOD
0 = PATH B BAD
1 = PATH B GOOD
PATH STATUS
DFREEQ INHIBIT
DISCARD INCOMING PACKETS WHICH
WOULD USE THE DFREEQ ENTRIES
INHIBITS DATAGRAMS OR DECNET MESSAGES
NR = RECEIVING SEQUENCE NUMBER
NS = SENDING SEQUENCE NUMBER
CIRCUIT STATE
0 = CLOSED
1 = OPENED (VIRTUAL CIRCUIT EXISTS)
TK-10649

Virtual Circuit Descriptor Table Entry

THE SBI INTERFACE

THE CMI INTERFACE

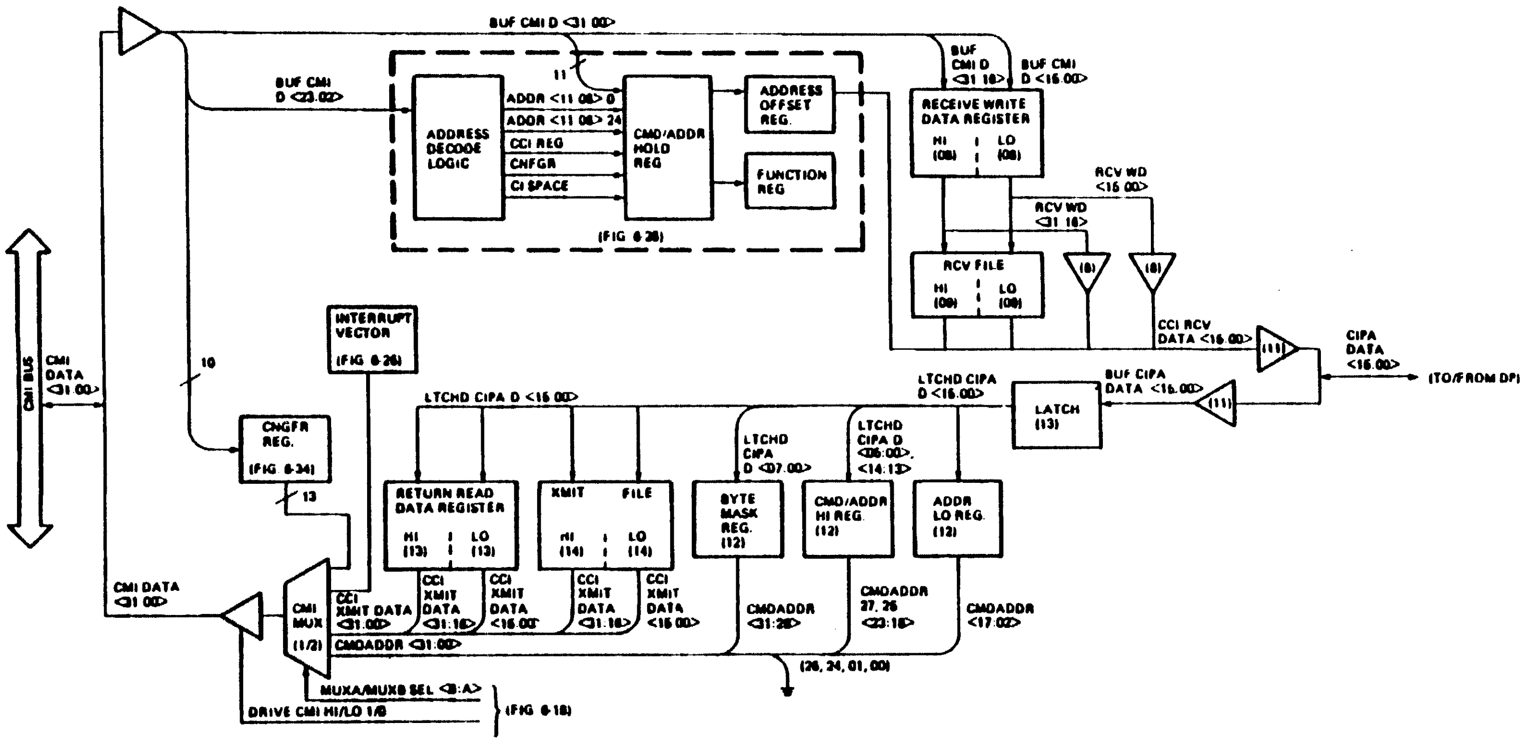
THE BICA INTERFACE



RD DATA T.O.

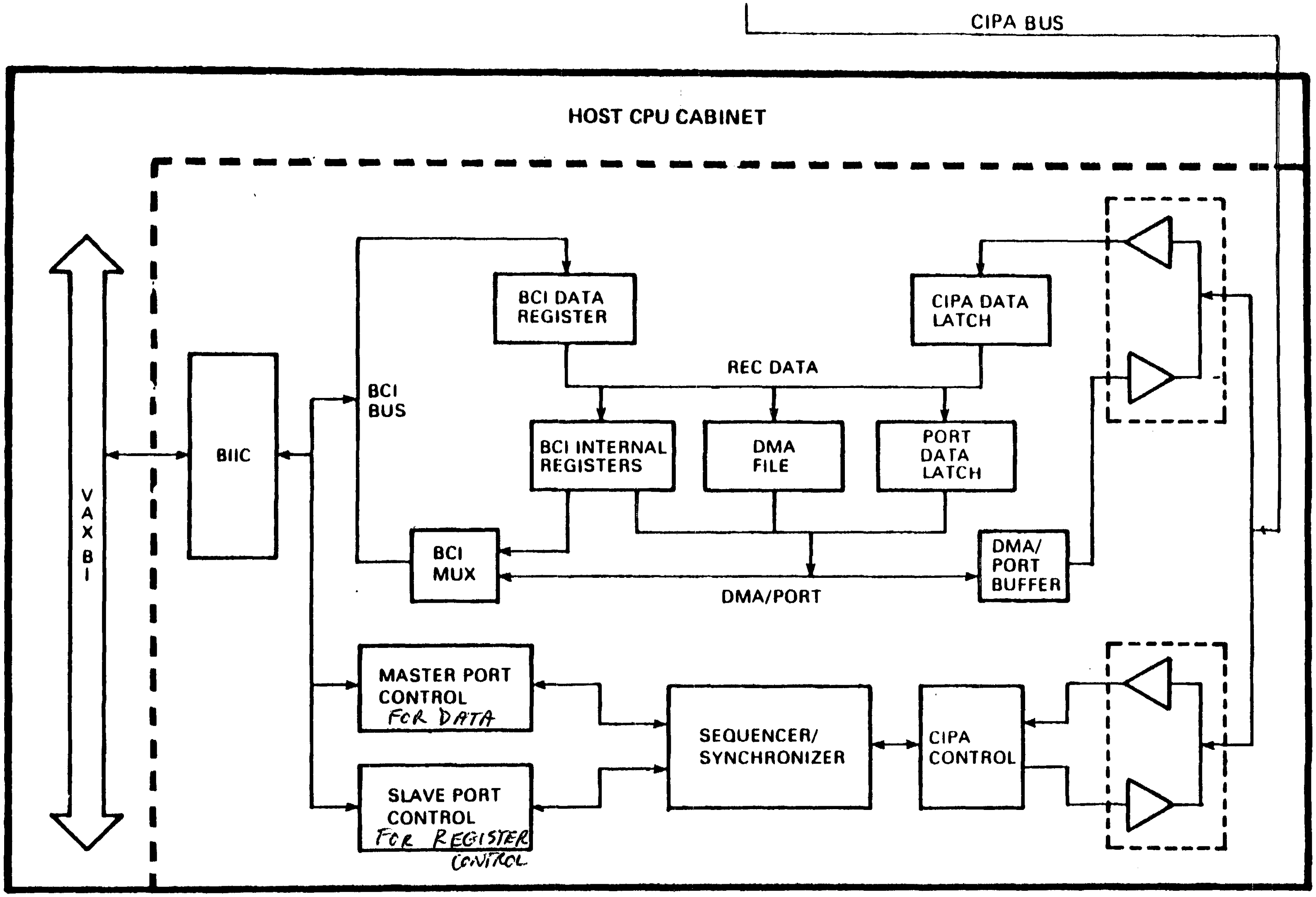
SBI PARITY ERRORS

SBI Module Block Diagram



NOTE
NUMBER DESIGNATIONS IN PARENTHESES
REFER TO ENGINEERING DRAWINGS
CONTAINING CORRESPONDING LOGIC

CI750 Interface Block Diagram (CCI)



CIBCI Interface (BICA Modules)

DKUTIL AND VERIFY UTILITIES

The purpose of this lab is to give you some exposure to two utilities that can be used to show the state of any disk connnected to the HSC. This lab should also help reenforce your understanding of DSDF (Digitial Standard Disk Format) and BBR (Bad Block Replacement).

✓ 1.) Run VERIFY on one of the disks attached to your HSC. Answer "no" to the question "Print informational (non-warning) messages?".

___ 2.) Take a look at the print-out and answer the following questions:

a.) How many physical blocks (PBNs) were found bad at the factory?__________

b.) How many bad LBNs were listed in the RCT? ________

c.) Will the contents of the FCT ever be larger than that of the RCT? ______

d.) Pick any PBN from the FCT and TRY TO! find it in the RCT.

e.) Pick an LBN from the RCT and find out to which cylinder and head it belongs by using the formulas given below:

$$\text{CYLINDER} = \frac{\text{LBN}}{\text{BPC}} \quad \text{(discard fraction)}$$

$$\text{HEAD} = \frac{\text{LBN} - (\text{CYL*BPC})}{\text{BPT}} \quad \text{(discard fraction)}$$

	BPC	BPT
	(Blocks/cylinder)	(Blocks/track)
RA60	252	42
RA80	434	31
RA81	714	51
RA82	855	57

f.) If the FCT were "null", what part of this print-out would be missing? ____________________________

g.) Looking at the RCT, how can you identify any non-primary revectors.

3.) Rerun VERIFY, but this time answer "yes" to the question "Print informational (nonwarning) messages?" and "yes" to the question "report transient errors by blocks?".

Notice the difference between this run and the previous one.

a.) Would a "transient" error be eliminated by running the formatter? ______________________________

b.) Do bad RBNs get revectored? ______
If not, what happens to them? ____________________
____

4.) Now lets use DKUTIL to find-out more information about the disk. Use the DISPLAY command to get the drive's characteristics.

a.) What is the drive's ECC threshold? _____

b.) How many sectors are there in one track? _____

c.) How many physical sectors are there on the whole disk? _____

5.) Using the DUMP command, pick an LBN from the RCT (get this off your verify printout) and dump it to the console.

a.) Locate the header, data, EDC and ECC areas.

b.) Can you find the header code?

6.) Now use the REVECTOR command to force a bad block replacement. Use one of the previously dumped LBNs.

a.) Does the revector command work for any LBN? ____

b.) Does the revector command work for an RBN? ___

c.) How would you find-out the new location of the block (blocks) you just revectored?

____

____

(try using the SET ERROR INFO command and see if a print-out will occur upon using the REVECTOR command, this should tell you the new LBN.)

SETSHO UTILITY

The SETSHO utility is used to examine/change internal operating paramaters of the HSC. Information on how to use it is in chapter 6 of the HSC User Guide, not the HSC70 Service Manual. A summary of all the available commands is on page 6-5 of the HSC User Guide.

___ 1. What does SET HOST do? ______________________________

___ 2. What does SET <unit id> HOST_ACCESS do?____________________

____

___ 3. Can the two above-named commands be used to dedicate one disk to one node? ______________________________

___ 4. What does SHOW MEMORY display? ___________________________

___ 5. When would you use the SET MEMORY ENABLE ALL command? ____

____

___ 6. What commands would you use to enable periodic and automatic diags? ____

____

___ 7. Does SHOW CI tell you what nodes are out on the cluster?

____

___ 8. Which command can you use to see what nodes are seen by the HSC CI manager software? _________________________________

___ 9. The contents of the errorlog on every host connected to the HSC can be partially controlled by what SETSHO command?

____

___ 10.The type of information printed-out on the HSC console is controlled by what SETSHO command? __________________________

___ 11. What command would you use to get a snapshot of all the requestors in the HSC?______________________________________

___ 12. Under the SETSHO prompt, type SHO SYS and refer to page 6-78 (HSC User Guide) for an explanation of all the fields displayed.

___ 13. If you found that you were unable to connect to the HSC from a remote terminal using DUP, what SETSHO command would help you with this problem? ___________________________________

___ 14. From a cluster perspective, which SETSHO parameters would consider the most important for proper operation of the cluster as a whole?

____

____

____

___ 15. Which SETSHO paramater is very important if you are going to dual-port drives between two HSCs? ______________________

___ 16. What setting would you recommend for the DUMP/NODUMP parameter? ______________________________________

___ 17. Try changing some of these parameters found in the HSC User Guide. Changing some of these require a reboot of the HSC. What page of the User Guide tells you which parameters will force an HSC reboot when you exit the SETSHO utility?

____

HSC Offline Diagnostics

THE STUDENT WILL FIND CHAPTER 4 OF THE HSC50 MANUAL AND CHAPTER 6 OF THE HSC70 MANUAL VERY HELPFULL.

___ 1. Load the OFFLINE diagnostic tape or floppy disk into the drive.

___ 2. Press the init button on the operator control panel.

___ 3. The offline diagnostic loader will go through a series of tests then display the offline diagnostic prompt:

```
ODL>
```

___ 4. Under the ODL> you can do a group of commands. They are found on pages 4-8 and 4-9 of the HSC50 Service manual and page 6-22 of the HSC70 manual. The first command you should try would be help. So type HELP after the ODL> prompt.

___ 5. Another usefull command would be SIZE. Go ahead try it. This command sizes up your system and shows you what is available to test.

How many requesters do you have in your HSC?

____

What are they?

____

What slots are they in?

____

What is the Requestor number for the K.CI?

____

What size is your Control Memory?

____

HSC OFFLINE DIAGNOSTIC LAB

INTRODUCTION

The purpose of this lab is to familiarize the student with the HSC's Offline diagnostics, their setup and operation.

OBJECTIVE

At the completion of this lab, the student will be able to:

- Set up the HSC to run the Offline diagnostics
- Use all Offline diagnostic loader commands
- Run all Offline diagnostics
- How to interpret all HSC Offline diagnostics error printouts
- Isolate a failing HSC module

REFERENCE MATERIAL

- HSC User Guide
- HSC50/70 Service Manuals
- Student Guide

What is the size of Data Memory?

____

What is the size of Program Memory?

____

___ 6. Under the ODL> prompt there are a limited amount of test commands. Using your manual, lets try running these tests. In the service manual, there is an error summary report after each test description. Use this to interpret any errors that might occur. The HSC Offline tests are:

___ TEST MEMORY (PG 6-73 HSC70 service manual)

How long would it take for a complete pass through Program memory?

____

What number would you use to specify Data memory?

___ (0) ___ (1) ___ (2)

What do you have to remember about the default parameters?

____

What would cause an error number #2?

____

What would cause an error number #12?

____

What is test number 2?

____

VTDPY LAB

1. Run VTDPY on the HSC and answer the following questions.

a. How many disks are on the HSC and what are their #s and status?

____

b. How many VAXes are communicating with this HSC and who are they?

____

c. How many tape drives are on the system and what is their status?

____

d. What processes are running on the system and how much CPU time?

____

____

___ TEST MEMORY BY K (PG 6-57 HSC70 manual)

Can this test be used to test Program memory?

___ YES ___ NO

Can you use requestor 0 for this test?

___ YES ___ NO

What number do you use to test Control memory?

___ (0) ___ (1) ___ (2)

How many passes could you specify for this diagnostic?

____

Error numbers 000 - 007 specify what memory?

___ Data ___ Control ___ Program

What does error number 11 specify?

____

How many tests make up this diagnostic?

____

___ TEST K (PG 6-43 HSC70 manual)

What tests are legal for a K.SDI?

____

What tests are legal for a K.CI?

____

Where do these tests come from?

____

What would an error number 9 indicate?

____

Unexpected traps would give me error numbers ___ or ___

Test #11 for the K.CI would test what board?

___ K.PLI ___ PACKET BUFFER ___ LINK

What test number tests the DATA BUS on the K.SDI?

___ (0) ___ (2) ___ (6) ___ (10)

What test number tests the SERDES on the K.STI?

___ (0) ___ (3) ___ (7) ___ (11)

___ TEST BUS (PG 6-33 HSC70 manual)

How many working Requestors are required for this diagnostic?

__ (1) ___ (2) ___ (4) ___ (none)

What memories could you use in this diagnostic?

___ Data ___ Control ___ Program ___ ALL

Could you interact with the Load device?

___ YES ___ NO

What would an error number 1 indicate?

____

What error numbers are specific to the HSC50 only?

____

What would an error number 15 for a HSC70 mean?

____

___ TEST REFRESH (PG 6-100 HSC70 manual)

How many patterns are used to test each memory bit?

___ (1) ___ (3) ___ (5)

What memories are tested by this diagnostic?

___ Data ___ Control ___ Program ___ All

What does error number 4 indicate?

____

What is the pattern used in test #2?

___ (000000) ___ (17777777) ___ (10101000)

___ TEST OCP (PG 6-104 HSC70 manual)

What does error number 1 mean?

____

What does test #3 test?

___ Init switch ___ Online switch ___ Fault switch

___ 7. Also under the ODL> prompt there are two commands called EXAMINE and DEPOSIT, that will allow you to examine and deposit into locations ir the HSC. Appendix A of the HSC50 service manual list some of these locations. Lets try using some of these.

___ ODL>E 17770056 This is the Serial number register (of the P.ioc)

___ ODL>E 17770046 This is the K Status register

___ ODL>E 17770042 This is the Switch register

___ ODL>E 17777564 This is the Transmit Status register for the console

Here is one, type BREAK key on console keyboard. At the @ sign type

What happens?____

__ 8. You can also do LOAD and START commands. Try some under the ODL> prompt do a LOAD <file name> where the file name is any program on the Diagnostic cassette or floppy disk. You can find out what is on the tape or floppy disk by doing a directory command under the HSC> prompt.

EXAMPLE: HSC50>DIR DD1: with the diagnostic tape in drive 1.

EXAMPLE: HSC70>DIR DX0:

After you load the program do the START command to run it.

How many tape patterns do you have?

What is Data pattern #3

What is Data pattern #17?

Up to how many minutes could you run this diagnostic?

What is Error #7 of this diagnostic mean?

What is Error #12 of this diagnostic mean?

What does Disk Error #102 mean?

What does Tape Error # 201 mean?

What does test #6 do?

What does test #10 do?

___ ILRX33 PG 5-2 HSC70 service manual [HSC70 only]

Does this test require a scratch diskette?

___ YES ___ NO

What does Error #3 mean?

____

How many tests make up this diagnostic?

____

___ ILTU58 PG 4-7 HSC50 service manual [HSC50 ONLY]

Does this test require a scratch cassette?

___ YES ___ NO

HSC50/70 INLINE Diagnostics

THE STUDENT WILL FIND CHAPTER 5 IN THE HSC70 MANUAL OR CHAPTER 4 OF THE HSC50 MANUAL HELPFUL.

INLINE DIAGNOSTICS ARE DIAGNOSTICS THAT WILL RUN WHILE THE HSC IS BOOTED UP AND RUNNING CLUSTER SOFTWARE OR AS A STANDALONE UNIT.

INLINE DIAGNOSTICS RUN UNDER THE HSC> PROMPT. THE DEVICE THAT YOU ARE TESTING MUST NOT BE ONLINE TO ANY CLUSTER MEMBERS. IF IT IS, IT MUST BE DISMOUNTED FROM THAT NODE.

1. The first operation is to bring up the HSC> prompt, that is done by booting up the system cassette or floppy disk. When it boots do a "CONTROL Y" to bring up the HSC> PROMPT. The INLINE diagnostics will be found on the System tape. You can do a directory command to to see them.

   To start a test just type RUN <device name>:<test name> and answer the questions.

```
        EXAMPLE  HSC70>RUN DX0:ILMEMY   for memory test on HSC70

                 HSC50>RUN DD0:ILMEMY   for memory test on HSC50
```

2. Lets start to run all the INLINE diagnostics for practice. In the HSC70 manual there is an Appendix [D] on Interpretation of status bytes that you can use to find any failures. Also chapter 5 after each test printout.

___ ILMEMY PG 5-6 HSC70 service manual

What user supplied parameters are needed for this test?

____

Exactly what part of memory does ILMEMY test?

____

___ ILDISK PG 5-9 HSC70 service manual

The INLINE disk diagnostic runs on what cylinders?

____

What prefix is used before the disk drive number?

What are the three FRUs that are tested using this diagnostic?

____

Why could would you receive an Error code of 03 for this test?

____

What would an Error 07 mean for this diagnostic?

____

What would an Error 30 mean for this diagnostic?

____

What does Test 7 do under the ILDISK diagnostic?

____

What does Test 12 do under the ILDISK diagnostic?

____

___ ILTAPE PG 5-31 HSC70 service manual

What is the prefix used before the drive number for this test?

____

What number would you use to select a density of 6250?

____

How many Data patterns do you have?

____

What does an Error 2 mean for this diagnostic?

____

HSC INLINE DIAGNOSTIC LAB

INTRODUCTION

The purpose of the INLINE diagnostic lab, is to familiarize the student with the setup and operation of the INLINE diagnostics for the HSC.

OBJECTIVE

After completing this lab, the student will be able to:

Set up HSC devices to run INLINE diagnostics

Run all INLINE diagnostics

Interpret INLINE error printouts

Isolate HSC failures from storage device failures

REFERENCE MATERIALS

HSC50/70 Service Manuals

HSC User's Güide

Student Guide

What does an Error 14 mean for this diagnostic?

____

What does an Error 31 mean for this diagnostic?

____

Name the three FRUs this diagnostic tests?

____

___ ILTCOM PG 5-47 HSC70 service manual

What is the purpose of this diagnostic?

____

What is an error 2 for this diagnostic?

____

What is an Error 5 for this diagnostic?

____

___ ILEXER PG 5-51 HSC70 service manual

Can you run this diagnostic through DUP?

___ YES ___ NO

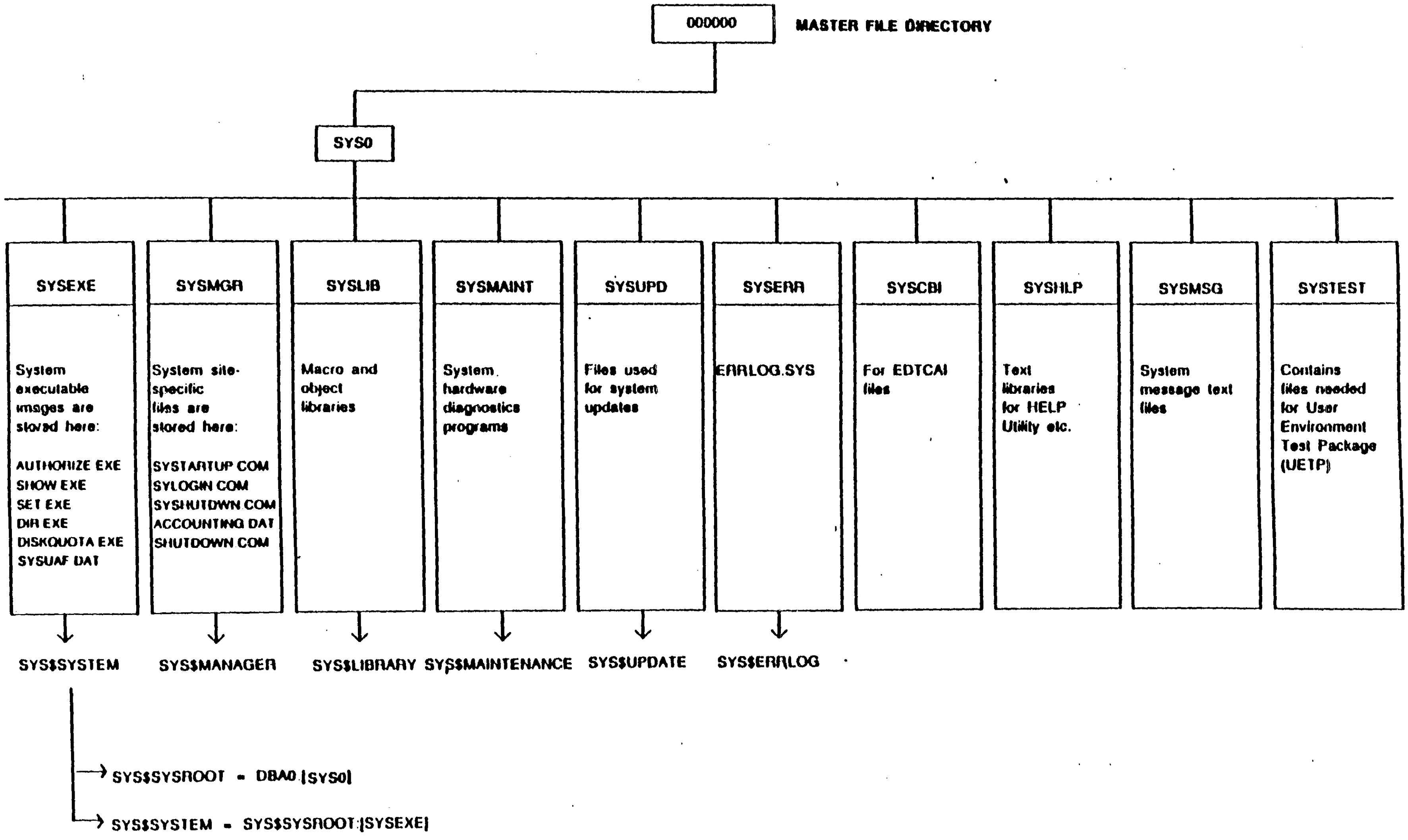
Can you use this diagnostic to access user areas?

___ YES ___ NO

How many disk Data patterns do you have?

____

SYSTEM DIRECTORIES ON DISK



000000 MASTER FILE DIRECTORY

SYS0

SYSEXE
System executable images are stored here:
AUTHORIZE EXE
SHOW EXE
SET EXE
DIR EXE
DISKQUOTA EXE
SYSUAF DAT
→ SYS$SYSTEM

SYSMGR
System site-specific files are stored here:
SYSTARTUP COM
SYLOGIN COM
SYSHUTDOWN COM
ACCOUNTING DAT
SHUTDOWN COM
→ SYS$MANAGER

SYSLIB
Macro and object libraries
→ SYS$LIBRARY

SYSMAINT
System hardware diagnostics programs
→ SYS$MAINTENANCE

SYSUPD
Files used for system updates
→ SYS$UPDATE

SYSERR
ERRLOG.SYS
→ SYS$ERRLOG

SYSCBI
For EDTCAI files

SYSHLP
Text libraries for HELP Utility etc.

SYSMSG
System message text files

SYSTEST
Contains files needed for User Environment Test Package (UETP)

→ SYS$SYSROOT = DBA0:[SYS0]

→ SYS$SYSTEM = SYS$SYSROOT:[SYSEXE]

117694

REGISTER QUIZ

1. WHICH 2 REGISTERS ENABLE ACCESS TO THE CONTROL STORE?

   -MADR MPATR

2. WHICH 2 HARDWARE REGISTERS ARE USED FOR ERROR REPORTING?

   PMCSR CIV FGF PESCR

3. IF BIT 3 (MAINT INTERUPT FLAG) IS SET IN THE PMCSR, WHAT ELSE SHOULD YOU CHECK?

   PSR is valid.

4. WHICH REGISTER IS USED TO START THE CI MICROPROCESSOR?

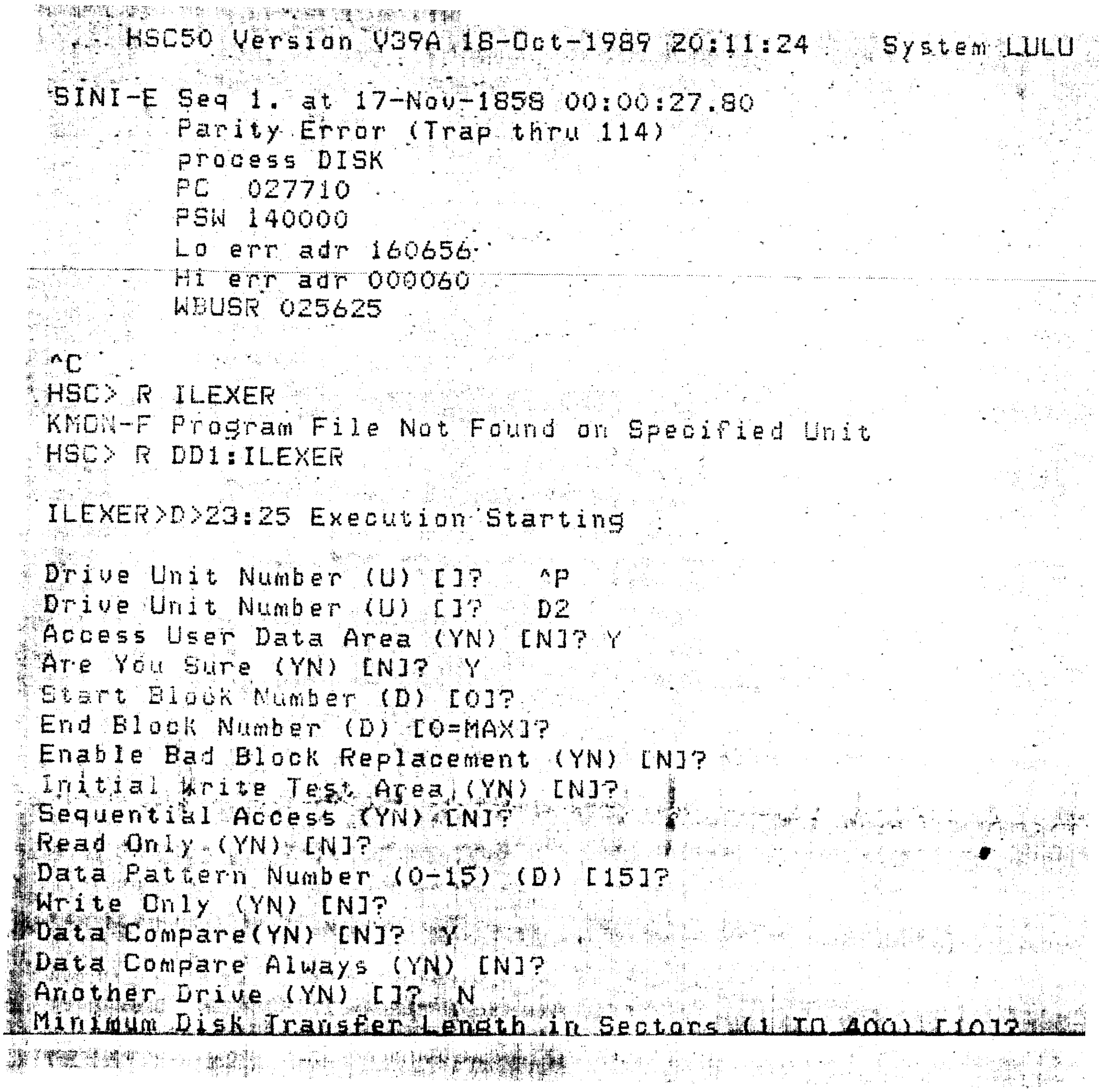
   PICR Bit 0

5. WHICH REGISTER IS USED TO STOP THE CI MICROPROCESSOR?

   PMCSR Bit 0

6. HOW DO MICROCODE REGISTERS DIFFER FROM HARDWARE REGISTERS?

   microcode Reg ARE not valid



```
  HSC50 Version V39A 18-Oct-1989 20:11:24      System LULU

 SINI-E Seq 1. at 17-Nov-1858 00:00:27.80
           Parity Error (Trap thru 114)
           process DISK
           PC  027710
           PSW 140000
           Lo err adr 160656
           Hi err adr 000060
           WBUSR 025625

 ^C
 HSC> R ILEXER
 KMON-F Program File Not Found on Specified Unit
 HSC> R DD1:ILEXER

 ILEXER>D>23:25 Execution Starting

 Drive Unit Number (U) []?    ^P
 Drive Unit Number (U) []?    D2
 Access User Data Area (YN) [N]? Y
 Are You Sure (YN) [N]?  Y
 Start Block Number (D) [0]?
 End Block Number (D) [0=MAX]?
 Enable Bad Block Replacement (YN) [N]?
 Initial Write Test Area (YN) [N]?
 Sequential Access (YN) [N]?
 Read Only (YN) [N]?
 Data Pattern Number (0-15) (D) [15]?
 Write Only (YN) [N]?
 Data Compare(YN) [N]?  Y
 Data Compare Always (YN) [N]?
 Another Drive (YN) []?  N
 Minimum Disk Transfer Length in Sectors (1 TO 400) [10]?
```

EV XXX
EVERY VAX

CI DIAGNOSTIC QUIZ

1. WHAT IS THE NAME OF THE DIAGNOSTIC TO TEST:

   A. CCI MODULE CNFGR REGISTER? ECCGA ~~EVCGB~~ WRONG ??

   B. 2911 MICROSEQUENCER ON THE CI780? EVCGD

   C. THE CI CABLES ON THE CIBCI? EVCKG

   D. COMMUNICATIONS BETWEEN NODES? EVXCI

   E. TRANSMIT BUFFER ON THE CI780? EVGGC

   F. CIBCA MICROCODE? EVGAA

   G. TO RELOAD THE CIBCA-BA MICROCODE?

      Run EVGAA or Run EVGDA
      set Event 1 update Eproms

uses Existing µ code

2. WHAT DIAGNOSTIC SHOULD YOU RUN TO TEST THE VIRTUAL CIRCUIT TO AN INTERMITTANT HSC? WHERE AND HOW SHOULD YOU RUN IT ??

   Level '3 EVGAA set flag 3

3. IF YOU'RE RUNNING EVCGD ON A BUSY CLUSTER AND GETTING ERRORS ON THE EXTERNAL LOOP TESTS, WHAT SHOULD YOU TRY ?

   Set flag 10 pg. 4-17

   Set Event 10

INTRO TO CLUSTER AND CI QUIZ

1. WHAT ARE THE MAJOR DIFFERENCES BETWEEN A CLUSTER AND A NETWORK ?

Cluster - Tightly coupled   Network - Loosley Coupled

2. WHICH PIECE OF HARDWARE IS THE HUB OF THE CLUSTER ?

STAR Coupler

3. WHICH PIECE OF SOFTWARE IS RESPONSIBLE FOR INTERNODE COMMUNICATIONS

Software communication services

4. WHICH PIECE OF SOFTWARE ENABLES QUEUE RESOURCE SHARING ?

Distributed JOb Controller

5. WHAT IS THE DIFFERENCE BETWEEN DESTINATION AND COMPLIMENT ?

Switch settings

6. WHAT IS CONTAINED IN THE BODY OF AN ACK PACKET ?

Nothing Body in Ack Packet Body

7. WHAT IS THE DIFFERENCE BETWEEN A NACK AND A NORESPONSE ?

NAck - unsuccesful reception because buffer is full

NO_RSP - unsuccessful reception becuase of collision, crc error, and so forth.

8. WHAT IS THE DIFFERENCE BETWEEN A VIRTUAL CIRCUIT AND A SOFTWARE CONNECTION ?

virtual - μcode in controllers port drivers logical connection between two nodes system not booted

Software - software drivers used for connections between nodes + disks

9. LIST THE 3 TYPES OF INFORMATION PACKETS AND WHAT EACH ARE USED FOR.

A. seq [illegible] transfer [illegible]

B. ~~DATA~~ gram

C. Block Transfer

> [illegible] 2-10

10. WHAT CI IS USED IN THE VAX 86XX AND WHY ?

CI780

11. WHAT IS THE CIPA BOX USED FOR ?

contains - DP module / PB module / Link module

12. NAME THE 4 MODULES THAT MAKE UP A CI780 (INCLUDE ENGINEERING TERMS)

SBI Interface module L0104 / Link Interface module L0100

DATA PATH module L0102

Packet Buffer module L0101

13. WHAT'S THE DIFFERENCE BETWEEN A NODE ADDRESS PLUG AND A NODE SWITCH

CI node number.

14. WHAT'S THE DIFFERENCE BETWEEN THE CIBCA-AA AND CIBCA-BA AND WHAT ARE THEIR MODULE NUMBERS ?

BA - microcode from Rom

AA - microcode from console media

HSC QUIZ 1

1. HOW MANY RA DISKS CAN EXIST ON AN HSC50/70/40 ?

   70 50

   32 / 24 | 12

2. HOW MANY DC POWER SUPPLIES EXIST IN AN HSC50/70/40 ?

   2 in 70 HSC 70 always has both supplies.

3. HOW MANY DISK/TAPE REQUESTORS CAN YOU HAVE ON AN HSC50 BEFORE YOU NEED MORE DC POWER ? 3

4. WHAT ARE THE 2 BLANK INDICATOR ON THE OCP USED FOR ?

   Fualt Display

5. HOW IS POWER REMOVED FROM THE CARD CAGE IN ORDER TO SWAP MODULES ?

   D.C. Power Switch 1-11

6. LIST ALL THE DIFFERENCES BETWEEN THE HSC50 AND THE HSC70.

   50 see page 1-9

7. WHAT IS THE TOTAL NUMBER OF MODULESTHAT CAN EXIST IN EACH HSC ?

50	40	70
11	8	13

8. WHAT IS CRONIC ?

   HSC O.S.

9. WHAT REQUESTOR NUMBER WILL THE K.CI ALWAYS BE ?

   1

10. WHERE IS THE -5.2 VOLTAGE CHECKED ?

    Back of P.S. +V2 P.OC P.OJ

11. WHERE IS THE PROGRAM MEMORY FOR THE P.IOJ ?

    1.2.6 HSC I/O Private Program memory

12. WHICH MEMORIES CAN THE K.PLI ACCESS ?

    1-3.6 control memory }
    DATA memory }

13. WHY DO THE K'S HAVE DUAL MICROSEQUENCERS ?

    one for control
    " " DATA

14. IN WHAT ORDER DO THE YELLOW LEDS ON THE P.IOJ TURN OFF ?

2-13 Vol 2 1, 4, 3, 2, 5

15. WHAT TESTS ARE IN THE ROM SEQUENCE FOR THE HSC70 ?

Init P.IOJ 3-15

INIPIO

16. WHAT LED SHOULD BE LIT ON THE K.SDI AFTER BOOTING THE OFFLINE TAPE ?

red

17. IF YOU BOOT THE HSC WITH NO MEDIA IN THE DRIVE, WHAT FAULT WOULD YOU GET ?

Code 23 3-8

18. WHAT STEPS ARE NEEDED TO GET TO THE ODL> PROMPT ?

Diag tape push init

19. WHAT COMMAND INITS ALL THE REQUESTORS ?

size - Checks the bus

20. WHAT TESTS ARE USED TO DIAGNOSE PROGRAM MEMORY ?

Test mem Test refresh

21. WHAT ARE THE TWO WAYS TO EXIT uODT,...WHICH SHOULD YOU TRY FIRST ?

1) @ P - proceed

2) @173000 G

CI780 PORT REGISTER LAB FOR VAX8600/8650

INTRODUCTION

The purpose of this lab is to introduce the student with the use of the CI port registers using a VAX8600/8650.

OBJECTIVE

After completing this lab the student will be able to:

Identify all CI780 registers

Explain the use of all CI780 registers

Use the CI780 registers to modify Cluster operation

Use the CI780 registers to troubleshoot the CI780 port adapter

REFERENCE MATERIAL

CI780 User's Guide

Student Guide

CI780 Port Adapter Registers

Since the VAX8600/8650 can have two SBI buses, the CI780 can have two base address ranges. For SBIA0, the base address of the CI port is 2001C000 for SBIA1 the CI780 the base is 2201C000. These are the base numbers that all the register offsets will be added to.

The 8600 CPU in this lab only has one SBI adapter (in SBIA0) so the base address will be 2001C000.

1. Halt the CPU by doing a CTRL P.

2. Examine the CI port Configuration register:

   ```
   >>>E/P/L 2001C000
   ```

   What is its address? ____________________

   What is the CI adapter code? ____________________

   Is this code the same for CI780 and CI750? ________________

   What bit sets if the port detects an SBI parity error? ________

   What bit sets if we have Corrected Read Data? _______________

3. Examine the Port Maintenance and Status register.

   What is its address?_________________________

   What bit tells me if the microcode is running?_____________

   How can I stop the microcode?______________________________

   ____

   What bit is set for Even Parity? __________________________

   What bit do you set if you want to force the microcode to start at a location in the Port Maintenance register? _______________

4. Examine the Port Maintenance Address register.

   What is its address?____

   How can I tell it to read or write to the high segment of the 47 bit microcode?____

   What must I do before I can use this register?___ ___________

   ______________ ____

   Deposit a 400 into this register. What do you see in the LEDS on the Packet Buffer board ?__________________________________

Deposit FFFFFFFF into the Maintenance Data register (BASE + 18) and read it back.____________

Deposit a 1400 into the Maintenance Address Register. Why doesn't 1400 appear in the LEDs? ________________________________

Now deposit FFFFFFFF into this location using the Maintenance Data register. What does it read out when you examine this location?

____

Why?____

5. Examine the Port Status register.

   What is its address?_________________________________

   Where is this address located?________________________

   What bit tells me that the port initialization is complete?

   ____

6. Examine the Port Parameter register.

   What is its address?_________________________________

   What does the 3F8 in the register mean?________________

   What is its port number?_____________________________

   Does it match your port number?_______________

   What register and address would I use to find the failing memory address? ____

   What address is the Port Initialize Control register at? ________

   ____

   What bit is the only one used in this register? _____________

7. Because we have been using location 400 and others we have destroyed portions of the microcode in the ram. Before doing the next step reload the microcode into ram by running the functional diagnostic with event flag 1 set.

8. Halt the system again by doing a CTRL P to put you back in console mode.

9. Let's halt the microcode:

```
>>>D 2001C004 1
```

Examine this location, bit 7 should now be a one:

```
>>>E 2001C004
```

Let's set the Programmable starting bit to a one.

```
>>>D 2001C004 40
```

Now set the address 400 into the Maintenance Address register.

```
>>>D 2001C014 400
```

Now start the microcode by setting a bit into the Port Initialize Control Register.

```
>>>D 2001C924 1
```

Now check to see if the port microcode has started.

```
>>>E 2001C004
```

Bit seven should be clear (0)

10. Now try examining the CI registers under the Diagnostic Supervisor. Boot the supervisor off the RL02 andd run the functional diagnostic EVGAA with event flat 1 set.

11. Try to deposit into the Maintenance Address register with the microcode running:

```
DS>D 2001C014 422
```

What happens? ______________________________

Try to deposit into the Maintenance Data register with the microcode still running.

What happens? ______________________________

Stop the microcode and try again, watch the LEDs on the Packet Buffer board while you do this.

Now can you do deposits? ______________________

12. Deposit FFF into address 400 (starting address of microcode). Now try to run EVGAA without reloading the microcode (event flag 1 clear).

Can you explain what happens? ______________________________

CI780 DIAGNOSTIC LAB

INTRODUCTION

The purpose of this lab is to familiarize the student with the setup and use of the CI port diagnostics for CI780.

OBJECTIVE

At the completion of this lab, the student will be able to:

Set up the Diagnostic Supervisor to run CI port diagnostics.

Run all CI780 repair level diagnostics

Run all CI780 functional diagnostics

Understand how the cluster environment affects the running of the Diagnostic Supervisor

Isolate a failing module

REFERENCE MATERIAL

CI780 User's Guide

Student Guide

CI780 PORT ADAPTER DIAGNOSTICS

1. All of the diagnostics for the CI port are all level 3. This means that they all must be run offline under the Diagnostic Supervisor. Use pages 3-1 thru 3-3 of the CI780 User's Guide as reference.

2. After you bring up the DS> prompt, you have to attach the devices you want to test and select them.

```
ex. DS>ATTACH CI780 SBI PAA0 14 4 2
    DS>SEL PAA0
```

Or you could run the autosizer program EVSBA from the cassette or disk.

```
ex. DS>RUN EVSBA
```

After it is done, do your SELECT command.

3. After you have done the setup, you can now run the repair diagnostics There are four of these diagnostics EVCGA thru EVCGD. To run the diagnostic type from the DS> prompt RUN (diagnostic name).

```
ex. DS>RUN EVCGA
```

___ EVCGA

___ EVCGB

___ EVCGC

___ EVCGD

After you have run the diagnostics in the default mode try running th with different sections enabled. An example would be to run ECCGE wit the manual section.

```
ex. DS>RUN EVCGD/SECTION:MANUAL
```

Try this and other sections with the rest of the port repair level diagnostics. Remember some tests run only in the manaual mode. To find out what sections are valid type HELP (DIAG NAME) SECTIONS.

What sections are valid for EVCGA? ______________________________

____

What sections are valid for EVCGB? ______________________________

____

What sections are valid for EVCGC? ______________________________

____

What sections are valid for EVCGD? ______________________________

____

Try also to run the diagnostics with event flags. To find the event f
that are legal for each diagnostic type HELP (DIAG NAME) EVENT_FLAG).

What flags are legal for EVCGA? ______________________________

What flags are legal for EVCGB? ______________________________

What flags are legal for EVCGC? ______________________________

What flags are legal for EVCGD? ______________________________

. Now lets try to run the Functional diagnostics EVGAA and EVGAB. Remem
if you run the repair level diagnostics first you will have to set ev
flag 1 to bring CI780.bin into memory and place it in the ram sectior
of the Packet Buffer board. Since CI780.bin is not normally found in
field account you might have to set your load path or put it into the
field service account.

```
Type DS>LOAD EVGAA
Type from the DS> prompt SET EVENT 1
Now type DS>START
```

Also try setting event flag 2 and 3 and observe the print out.

Try running EVGAB now.

REMEMBER THESE DIAGNOSTICS ARE VERY DEPENDENT ON THE REV LEVELS OF
CI780.BIN AND DIAGNOSTICS.

PART 2: BOOTING DIAGNOSTIC SUPERVISOR FROM AN HSC DISK

› load the Diagnostics from the HSC disk type:

```
>>>@diaboo.cmd      This file was created to boot the Diagnostic
                    Supervisor from the HSC for this VAX.
```

Type the following:

```
DS> dir          [000000.VMS$COMMON.SYSMAINT]
DS> set load ~~[sys$common.sysmaint]~~
DS> dir
```

. Attach the CI780, or run the autosizer, then run EVCGA. Now try to run EVCGB. Try the DIR command.

What happens?____

____

Why? ____

What does this tell you about putting CI port diagnostics on the HSC

Disk?____

CLUSTER QUIZ

1. HOW IS QUORUM CALCULATED?

2. WHAT IS PARTITIONING AND WHY IS IT DANGEROUS?

3. HOW DOES THE VAX KNOW WHICH ROOT TO LOAD ITS OPERATING SYSTEM FROM?

4. WHAT DO R2 AND R3 DEFINE IN THE BOOT PROCESS?

5. HOW DO YOU TELL VMB.EXE THAT THE SYSTEM DISK IS DUAL PORTED, AND WHY WOULD YOU.

6. WHAT DOES REMOVE_NODE DO IN SHUTDOWN?

7. HOW IS CLUSTER_SHUTDOWN USED TO BRING THE WHOLE CLUSTER DOWN AND WHY?

8. WHAT WOULD A DEVICE NAME OF $215$DUA1: TELL YOU ABOUT THE DEVICE?

9. WHAT ARE THE STEPS (COMMANDS) REQUIRED TO SERVE AND MOUNT A DUAL PORTED DISK TO A CLUSTER?

10. HOW DOES VMB.EXE KNOW THAT THE SYSTEM DISK IS A MEMBER OF A SHADOW SET? HOW DOES IT KNOW WHICH SET?

11. WHEN WOULD YOU USE THE DIAG SUPERVISOR SET LOAD COMMAND IN A CLUSTER AND WHY?

12. WHY IS THE TOP LEVEL SYSMAINT DIRECTORY NOT USED ON A VMS_V5 SYSTEM?

13. WHAT TWO THINGS SHOULD YOU ALWAYS CHECK ON A CLUSTER BEFORE SHUTTING DOWN A NODE?

14. SYS$SYSROOT IS WHAT KIND OF LOGICAL NAME?

15. WHAT ARE THE EQUIVALENCY STRINGS FOR THE SEARCHLIST SYS$SYSROOT?

16. HOW CAN YOU GUARANTEE THE SHOW CLUSTER WILL NOT ONLY GIVE ME ALL THE INFORMATION YOU NEED BUT WILL DO SO WITH A SINGLE COMMAND?

17. WHAT SYSGEN PARAMETERS SET UP A QUORUM DISK?

18. IF A VAX SYSTEM WILL NOT BOOT THROUGH THE CI, BUT ALL DIAGS PASS, WHAT SHOULD YOU CHECK ON THE HSC?.....ON OTHER SYSTEMS?

19. IF QUORUM IS LOST ON A CLUSTER, WHAT ARE YOUR OPTIONS?
(LIST THEM IN PREFERED ORDER)
A.
B.
C.

20. IF AN NI NODE WIL NOT BOOT, BUT ALL DIAGS PASS, WHAT SHOULD YOU CHECK?

This is an example as to how to get into the system without knowing a PASSWORD...works for a VAX8200/8250/8300/8350.

Notice the fact that there are blank lines in between some of the different commands that you have to type in. The messages that the system will present to you, are NOT included.

```
>>> B/R5:1 DU40          ;; we assume that the system device is DU40.


SYSBOOT> SET/STARTUP=OPA0:
SYSBOOT> EXIT
$ SET NOON
$ SPAWN

$ @SYS$SYSTEM:STARTUP.COM

$ SET DEF SYS$SYSROOT:[SYSEXE]

$ RUN AUTHORIZE

UAF> MOD SYSTEM/PASS=8200MAINT

UAF> EXIT

$ ^P               ; halts the system abruptly, not recommended for normal
                   ; operation.

>>> H              ; This command is here to keep the VAX 11/780 "gurus"
                   ; satisfied, it is not needed.

>>> I

>>> B DU40


$ @SYS$SYSTEM:STARTUP.COM        ; after this command, wait a few
                                 ; seconds, then type <CR>


USERNAME: SYSTEM
PASSWORD: 8200MAINT
```

```
;; We must now set the system to point to the normal STARTUP
;; program.

   CR SYSGEN
SYSGEN> SET/STARTUP=SYS$SYSROOT:STARTUP.COM
SYSGEN> WRITE CURRENT
SYSGEN> EXIT
$ @SYS$SYSTEM:SHUTDOWN
<CR>
<CR>
<CR>
<CR>
<CR>
<CR>
<CR>
<CR>

;; Now the system has at least one known password.....have fun !!!
```

THAT'S ALL FOLKS!!.

The attached information is from the CSSE Mass storage west Group
It contains important HSC Controller information
(The following article can be found in the CSSE STARS database)

FROM: Ron Repka
DEPT: Maintainability Eng.
LOC.: CX01-2/Q12
TEL.: 303-548-6195/522-6195
ENET: SSDEVO::REPKA

SUBJECT: PERFORMANCE CONSIDERATIONS FOR HSC CONTROLLERS.

1 INTRODUCTION

There have always been performance considerations with regards to configuring devices on the HSC. Until recently, these considerations were important, but not extremely critical because the speed of the available devices did not tax the bandwidth of the HSC. With the introduction of the RA82, RA90, TA90, and RA70, configuring for maximum performance has become very important. In addition, if this is not done, not only will there be a effect in performance, but data buss contention caused errors may also occur, such as data buss overruns, drat seek timeouts, and EDC errors.

Some general rules can be given, but things such as the customer's application and the usage of the devices also need to be considered when looking at the overall configuration. This memo will try to provide some general guidelines and some understanding of what effects performance in the HSC and and to what degree.

There are several things to keep in mind when configuring a device on a HSC.

1. Only one device on a requestor can transfer data at one time.

2. Shadow set members need to be on separate requestors.

3. Requestor priority.

4. The speed of the device.

Sections 2, 3 and 4 will discuss these points in detail. Section 5 will offer some configuration guidelines, and section 6 discusses a performance issue specific to single ported drives.

2 ONLY ONE DEVICE ON A REQUESTOR CAN TRANSFER DATA AT A TIME.

Each requestor can support 4 devices, however, only one of these devices can transfer data at any one time. While one device is transferring data, other devices can be seeking or positioning tape. This means, the fewer fast and heavily used devices on any one requestor the better the performance.

There is no significant performance difference between individual ports within a requestor.

3 SHADOW SET MEMBERS MUST BE ON SEPARATE REQUESTORS.

This is a most important performance consideration. When the HSC picks which drive to read from on a shadow set, the first check made is for members on the same requestors. If all members are on the same requestor, the HSC just uses the primary member and does no read optimization at all. This means the same drive is always used on reads and the speed advantage of reads in a shadow set are lost. On writes, only one drive can be written to at any one time while simultaneous writes could occur if the drives were split across different requestors.

4 REQUESTOR PRIORITY AND DEVICE SPEED.

Requestor priority in the HSC is used two different ways. The first, and the most obvious, is if two requestors are contending for the data buss at the same time, the higher priority (higher numbered) requestor wins.

The second way requestor priority is used is not so obvious and requires some further explanation.

Each device connected to the HSC reports back it's transfer rate to the HSC when it is brought on line. The HSC uses this to allocate bandwidth on the data buss along with requestor priority. A RA90 needs every 3rd data buss cycle with it's high transfer speed so this means 3 RA90s can transfer data at the same time. The RA70 needs only every 5th cycle so 5 RA70s can transfer at the same time. This is assuming the drives are on separate requestors. The requestor module buffers less than 2 words, so if the RA90 cannot get the buss for 3 consecutive cycles, a data buss overrun will occur.

This works very well as long as the faster devices are at the higher requestor numbers. If a RA90 is running with 3 RA70s, the RA90 will always get the buss when it needs it as long as it is at a higher requestor number. However, if the RA90 is installed at a lower requestor number than the RA70s, it is possible that 3 RA70s will lock out the RA90 for 3 cycles an cause a overrun condition. To prevent this, the HSC checks the speed and priority of the devices, and if it detects a slower device at a higher priority, it allocates the slower device the same bandwidth as the faster device.

For example, if RA70s were placed higher than RA90s, the HSC would consider the RA70 the same speed as RA90s and would allow only 3 RA70s to transfer at the same time or any combination of 3 RA90s/RA70s. This ensures the RA90 will get at least the one out of every 3 cycles it needs. This also means 40% of the HSC bandwidth would not be utilized. For this reason, it is very important to configure devices according to their relative transfer speeds. The relative speeds of currently supported devices from the fastest to the slowest is:

TA90
RA90
RA82
RA81
RA60
RA70
RA80
TA78/79/81

TA78/79/81 tape drives, unlike disk drives, do not have steady transfer rates due to the irregularities of tape movement. Instead, their transfer rates are very erratic and bursty in nature. For this reason, they can cause bandwidth problems for other devices if given higher priority on the data buss, even though their "average" transfer rate is much slower. For this reason it is especially important to keep them at the lowest requestor numbers. This means they cannot be mixed on the same requestor with the cached TA90 which must be at the highest requestor numbers.

5 GUIDELINES

Based on the above information, some basic guidelines can be formulated to aid in configuring the HSC and it's devices. First, configure the devices of different device types according to the following criteria in order of priority;

1. TA78/79/81 tape drives must be at the lowest requestor numbers. TA90s must be at the highest requestor numbers and cannot share a requestor with TA78/79/81 type drives.

2. Shadow set members must be on separate requestors. This has the highest performance impact and the highest priority.

3. Faster devices should be installed on higher requestor numbers. This ensures the maximum possible transfer rate will be maintained and eliminates the possibility of data buss contention caused errors. Having shadow members on separate requestors may conflict with this rule, in which case, shadow members on different requestors takes priority.

4. Faster and high usage devices should be on requestors by themselves, or as few of these devices on any one requestor as possible.

Finally, after configuring the different device types using the above guidelines, look at the devices of the same type. When configuring devices of the same type, usage of each device needs to be considered along with requestor priority and the fact that a requestor can only transfer from one port at a time to determine the best configuration.

For example, the heaviest used device of one type, might be placed on a requestor by itself, or with a device that is seldom used and possibly placed at the highest requestor number of the devices of that same type. Some judgment will have to be used here along with some knowledge of the customer's application.

6 ADDITIONAL PERFORMANCE CONSIDERATION FOR SINGLE PORTED DISK DRIVES.

VMS version 5.0 increased the frequency of performing it's Determine Access Path processing (DAP). DAP processing is used by VMS to periodically cause the disk drives to release their ports so VMS can determine all access paths to all drives. What this translates to in the HSC/drive is a topology command being sent to the drive. This topology command causes the drive to release it's selected port and send an attention/available to the other port if the port button is selected. During this time the drive is unavailable for disk transfers, however, in a normal dual ported configuration it takes o[illegible] milliseconds for the attention/available and the HSC response.

The problem occurs when the drive has both port buttons selected and is only connected to one controller. In that case, the drive sends the attention/available on the unconnected port, and sits there and waits for a response until it eventually times out after about 2 seconds. This means all IO to the drive stalls for 2 seconds every time VMS calls DAP processing. Currently this will happen only every 6 minuets but VMS is expected to increase this frequency in the future.

Even though DAP processing is relatively infrequent now, it is important for the Customer and Field Service to understand that there still is some effect on performance by keeping both port buttons selected on single ported drives and this may be of larger concern in the future.

==================================== END ====================================